

# ADS 高速电路信号完整性应用实例

张 涛 等编著

電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

## 内 容 简 介

本书主要介绍使用 ADS 软件进行高速电路信号完整性设计的方法, 包含 13 个工程实例, 详细介绍了传输线阻抗分析、串扰分析、TDR 仿真、串行总线与 DDR 总线、电源完整性、仿真与测量结合的设计与分析方法。本书的特点是以工程实例为主, 结合理论分析, 实用性强。

本书适合从事射频电路设计与仿真的工程技术人员阅读使用, 也可作为高等院校相关专业的教学用书。

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。  
版权所有, 侵权必究。

## 图书在版编目 (CIP) 数据

ADS 高速电路信号完整性应用实例/张涛等编著. —北京: 电子工业出版社, 2016. 1  
ISBN 978-7-121-27392-6

I. ①A… II. ①张… III. ①电路设计—计算机辅助设计—软件包 IV. ①TN702

中国版本图书馆 CIP 数据核字 (2015) 第 243382 号

策划编辑: 张剑 (zhang@phei.com.cn)

责任编辑: 夏平飞

印 刷:

装 订:

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1092 1/16 印张: 16 字数: 406 千字

版 次: 2016 年 1 月第 1 版

印 次: 2016 年 1 月第 1 次印刷

印 数: 3 000 册 定价: 48.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。



# 前 言

数据传输速率的不断提高,使得信号完整性的问题变得越来越突出,借助于仿真工具进行信号完整性的分析和预测,已逐渐成为重要的设计手段。是德科技的 ADS 软件针对这一挑战提供了从通道建模到仿真分析再到测量验证的完整的解决方案,是目前市场上主流的高速电路信号完整性仿真平台。

在射频微波电路设计领域,ADS 软件已经被广大工程师了解和接受,国内也已有多部专著介绍了相关应用。但在高速电路信号完整性领域,目前的专著较少,希望本书的出现能够填补这一空白。

本书以案例为主,不包含信号完整性的基础理论知识。书中选择了高速电路设计领域的若干个典型案例,内容范围涉及传输线设计、连接器设计、高速背板建模与验证、串行及并行总线信号质量分析、电源完整性分析等多个领域。

本书包括 13 个实例及一个附录。其中,实例 1 和实例 2 由赵晨星编写,主要介绍传输线阻抗分析方法;实例 3 和实例 9 由谢成诚编写,实例 3 介绍连接器仿真与 TDR 分析方法,实例 9 介绍光通道 IBIS-AMI 建模及验证方法;实例 4 和实例 10 由薛新东编写,实例 4 介绍高速背板的建模与测量验证,实例 10 介绍从 Cadence 版图设计工具 Allegro 导出版图至 ADS 软件中进行仿真的基本流程;实例 5 和实例 8 由王熠编写,实例 5 介绍基于矢量网络分析仪测量的宽带 PCB 材料参数提取方法,实例 8 介绍如何将高速实时示波器测量的波形结合实际芯片的 IBIS-AMI 模型进行接收的均衡和后处理,分析芯片内眼图和信号质量;实例 6 和实例 7 由张涛编写,实例 6 介绍如何将测量的 CEI-25G-LR 通道数据根据协议定义的规范进行一致性分析,实例 7 介绍通道仿真器的原理、设置及应用;实例 11 和实例 12 由崔梦编写,介绍 DDR4 仿真器及一致性测试的基本方法;实例 13 由陈哲生编写,介绍使用 ADS 进行电源网络阻抗分析的基本流程。书中除实例 12 外,其余实例都使用 ADS2014.01 版本进行了验证,这些实例在更早的版本,如 ADS2012、ADS2013 也可以使用。

附录 A 中列举了 22 个常见的使用 ADS 软件进行高速电路信号完整性仿真问题,并逐一进行了解答。所选问题来自我的中国台湾同事林鸣志先生编写的《ADS 信号完整性问题集》,在此一并表示感谢。

由于编者水平有限,加上时间紧迫,书中错误在所难免,希望广大读者批评指正。

为便于读者阅读、学习,特提供本书实例下载资源,请访问 <http://yydz.phei.com.cn> 网站,到“资源下载”栏目下载。

编著者



# 目 录

实例 1 单端传输线阻抗分析 .....	1
1.1 创建 ADS 项目文件 .....	1
1.2 仿真电路建立 .....	3
1.3 电路优化 .....	8
1.4 版图仿真设定 .....	9
1.5 电磁仿真 .....	12
1.6 CILD 工具使用 .....	17
实例 2 差分传输线分析 .....	21
2.1 创建 ADS 项目文件 .....	21
2.2 差分线版图元件创建 .....	21
2.3 差分线损耗串扰仿真 .....	22
2.4 差分线阻抗分析 .....	24
2.5 单端 S 参数到差分/共模 S 参数的转换 .....	27
2.6 SnP 文件导入 .....	29
实例 3 传输线及三维连接器 TDR 仿真 .....	34
3.1 PCB TDR 仿真实验 .....	34
3.2 PCB 及连接器 TDR 仿真实验 .....	48
实例 4 基于测量的通道建模 .....	56
4.1 通道测量结果显示 .....	56
4.2 通道时域特性仿真 .....	57
4.3 建立通道模型 .....	59
4.4 通道模型的优化 .....	61
4.5 使用优化通道模型进行仿真 .....	66
实例 5 PCB 板材宽带参数提取 .....	68
5.1 测量文件的导入和验证 .....	71
5.2 AFR 和测量结果去嵌入 .....	73
5.3 介电常数拟合 .....	78
实例 6 CEI-25G-LR 通道特性一致性分析 .....	82
6.1 建立通道模型并仿真 .....	83
6.2 通过 AEL 脚本语言自定义测量函数 .....	84
6.3 插入损耗曲线 .....	85
6.4 拟合插入损耗曲线 .....	86
6.5 插入损耗曲线偏差 .....	88
6.6 回波损耗曲线 .....	89

6.7 综合串扰噪声 .....	90
<b>实例 7 高速串行链路的通道仿真 .....</b>	<b>92</b>
7.1 PCIE 通道模型 .....	104
7.2 基本的通道仿真 .....	105
7.3 串扰 (Xtalk) 对通道性能的影响 .....	106
7.4 CTLE 均衡 .....	106
7.5 FFE 均衡 .....	109
7.6 DFE 均衡 .....	110
7.7 通道的阶跃响应与冲击响应 .....	110
7.8 IBIS – AMI 模型仿真 .....	112
7.9 通道参数扫描与最优化 .....	113
<b>实例 8 使用 IBIS – AMI 模型分析高速串行链路实测波形 .....</b>	<b>116</b>
8.1 测量波形 .....	117
8.2 通道建模 .....	119
8.3 模型验证 .....	122
8.4 AMI 模型的后处理和分析 .....	128
8.5 扫描参数 .....	130
<b>实例 9 光通道 IBIS – AMI 模型创建及验证 .....</b>	<b>133</b>
9.1 在 SystemVue 中进行光通道 IBIS – AMI 模型创建 .....	134
9.2 在 ADS 中进行光通道 IBIS – AMI 模型验证 .....	140
<b>实例 10 Cadence Allegro 版图的提取、导入及仿真 .....</b>	<b>144</b>
10.1 AllegroDFI 插件的安装及设置 .....	144
10.2 在 Allegro 软件中使用 AllegroDFI 插件进行版图提取 .....	146
10.3 创建 ADS 工程, 导入 Allegro 版图 .....	153
10.4 查看导入的版图 .....	155
10.5 Momentum 设置及仿真 .....	160
10.6 将电磁场仿真结果代入 ADS 原理图进行仿真 .....	173
<b>实例 11 DDR4 仿真及一致性测试 .....</b>	<b>181</b>
11.1 DDR4 一致性测试关键指标 .....	191
11.2 DDR4 一致性测试步骤 .....	192
<b>实例 12 DDR BUS 仿真器 .....</b>	<b>198</b>
12.1 DDR4 一致性测试面临的挑战 .....	198
12.2 DDR BUS 仿真设置实例 .....	200
<b>实例 13 电源完整性分析——电源网络阻抗分析 .....</b>	<b>207</b>
13.1 版图的截取与导入 .....	207
13.2 将 ADFI 导出的版图导入到 ADS .....	210
13.3 EM setup 设置 .....	214
13.4 PDN 阻抗响应 .....	220
<b>附录 A ADS 信号完整性仿真常见问题及解答 .....</b>	<b>223</b>

# 实例 1 单端传输线阻抗分析

## 【目的】

本实例主要介绍单端传输线特性阻抗的分析方法。实例中将使用原理图微带模型、平面电磁场仿真模型以及 CILD（Controlled Impedance Line Designer）工具三种方法。



## 1.1 创建 ADS 项目文件

(1) 打开 ADS 主程序，在起始窗口（Getting Started with ADS），单击“Create a new workspace”，创建一个新的项目，如图 1-1 所示。

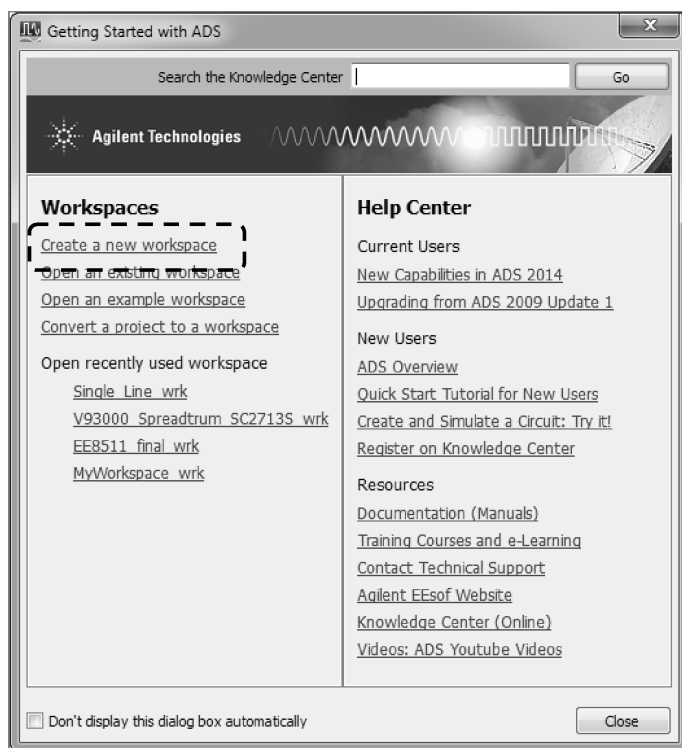


图 1-1 ADS 起始窗口

(2) 弹出新建项目向导（New Workspace Wizard），单击下一步。

(3) 在 Workspace name 栏目中输入新建项目的名称“Single\_Line\_wrk”，在 Create in 栏目中选择该项目所保存的路径及文件夹，单击下一步，如图 1-2 所示。

(4) 在“Add Libraries”窗口中，ADS 库选择 Analog/RF，如项目涉及数据流仿真，则

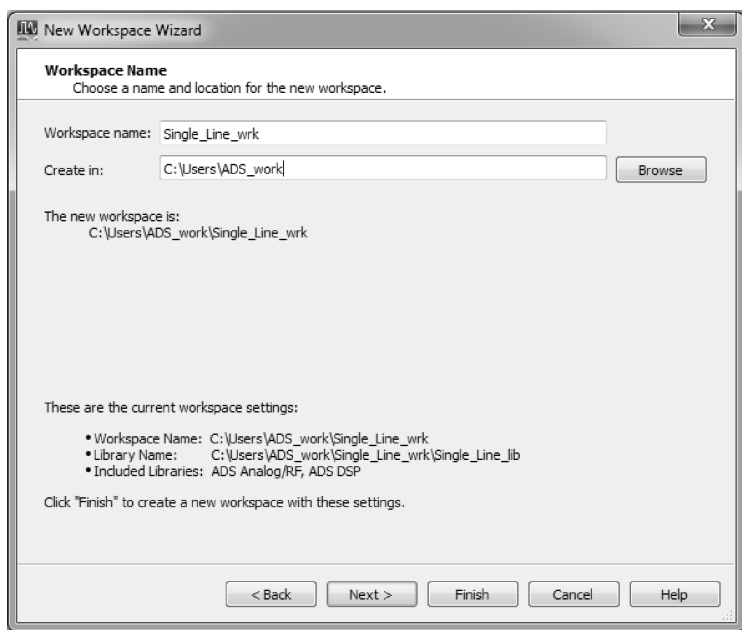


图 1-2 ADS 新建项目向导

需要勾选 DSP。如果需要添加厂商提供的元器件库和 PDK，可以在这个界面看到可供选择的库，根据需要勾选即可。单击下一步，如图 1-3 所示。

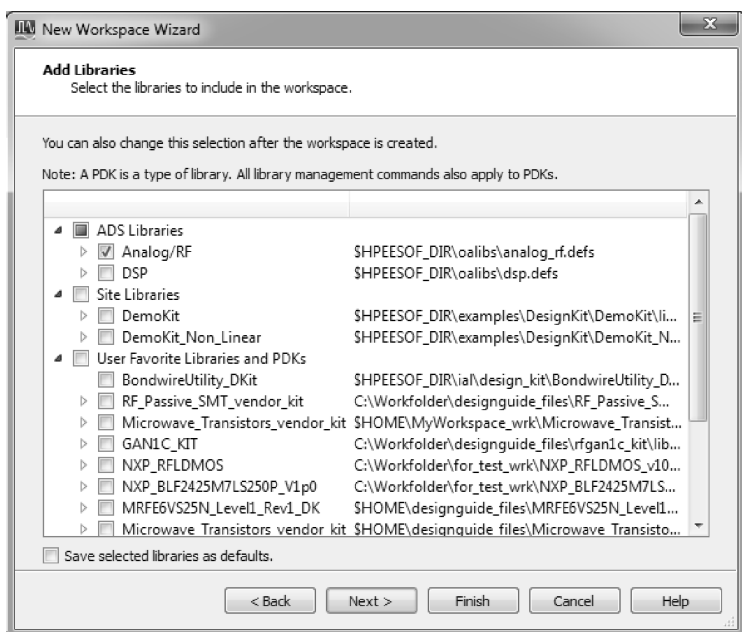


图 1-3 添加库窗口

(5) 在“Library Name”窗口中，推荐使用默认的库名，默认库名和项目名称相同，如图 1-4 所示。

(6) 在“Technology”窗口中可以设定该项目版图的单位、分辨率、ADS 标准层定义等

有关工艺的项目，此处设置如图 1-5 所示。

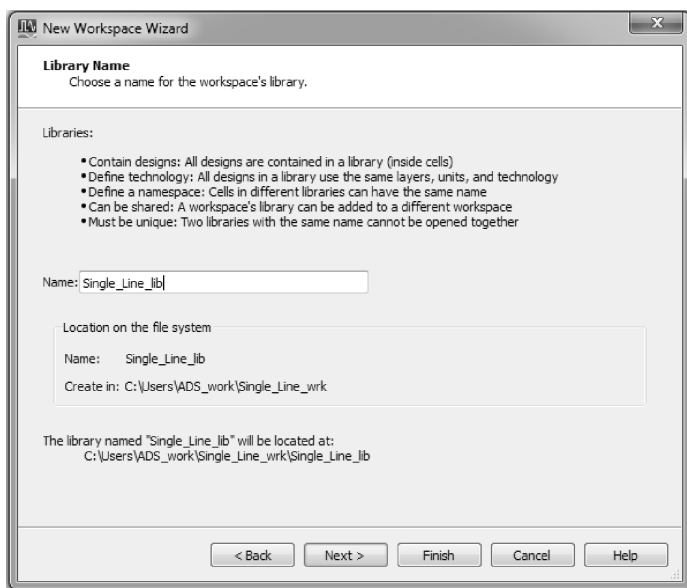


图 1-4 库命名窗口

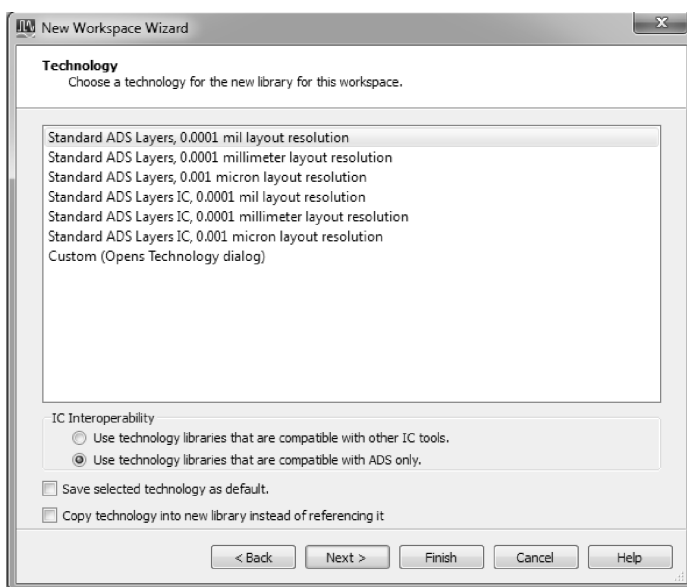


图 1-5 工艺定义窗口



## 1.2 仿真电路建立

(1) 在项目 Single\_Line\_wrk 下，使用“New Schematic window”建立一个新的原理图“Single\_Line”，如图 1-6 所示。在模板调用选项中使用“ads\_templates: S\_Params”，ADS 将

在电路中自动添加 S 参数仿真器以及输入和输出端口。

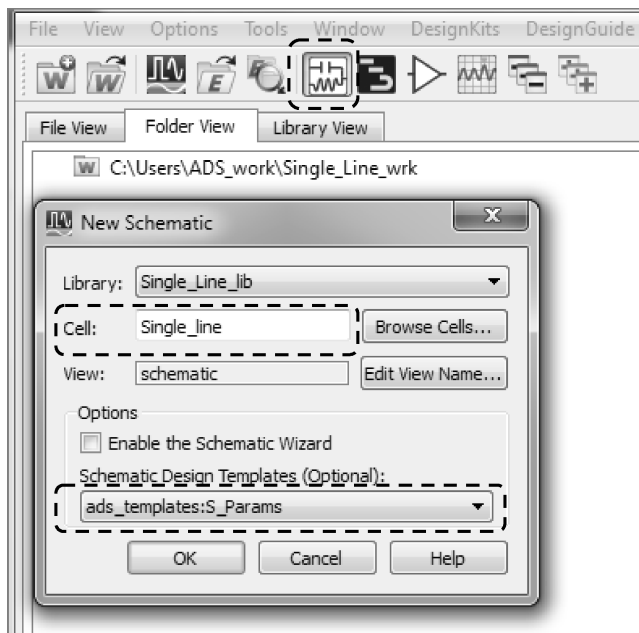


图 1-6 新建电路窗口

(2) 从原理图“TLines\_Multilayer”元件库中选择“ML1CTL\_C”和“MLSUBSTRATE2”元件，按照图 1-7 所示搭建电路，“MLSUBSTRATE2”各项关键参数的设定如下：

☺  $\epsilon_r = 4.5$

☺  $T[1] = T[2] = 1.7 \text{ mil}$

☺  $H = 10 \text{ mil}$

☺  $\text{Cond}[1] = \text{Cond}[2] = 5.8 \text{E} + 7$

☺  $\text{TanD} = 0.001$

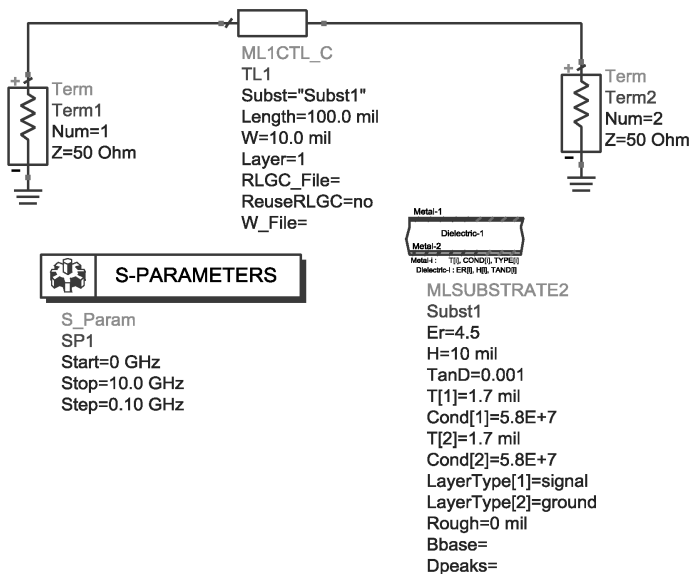


图 1-7 单端传输线电路



“ML1CTL\_C”参数设定如下：

☺ Length = 100.0mil      ☺ W = 10.0mil

(3) 本范例的主旨在于使用优化的办法来求得单端传输线的特性阻抗，基本原理是利用端口阻抗若是和传输线阻抗完全匹配的情况下，端口处的反射系数理论上会非常小。

从“Optim/Stat/DOE”元件库中选择 Optim 和 Goal 放置在电路中。Optim 和 Goal 分别为 ADS 中的优化和目标控件，用于电路的优化。可以通过设定一个或者多个目标，来对电路中的一些变量进行优化，优化的时候根据优化控件中优化方法等设定，对优化进行控制，从而达到优化电路的目的。

(4) 设定变量 Z0，如图 1-8 所示，Z0 的初始值为 50。在变量的设定选项中，将 Z0 设置为一个可优化的连续变量，设定优化的范围为 1 ~ 1000。

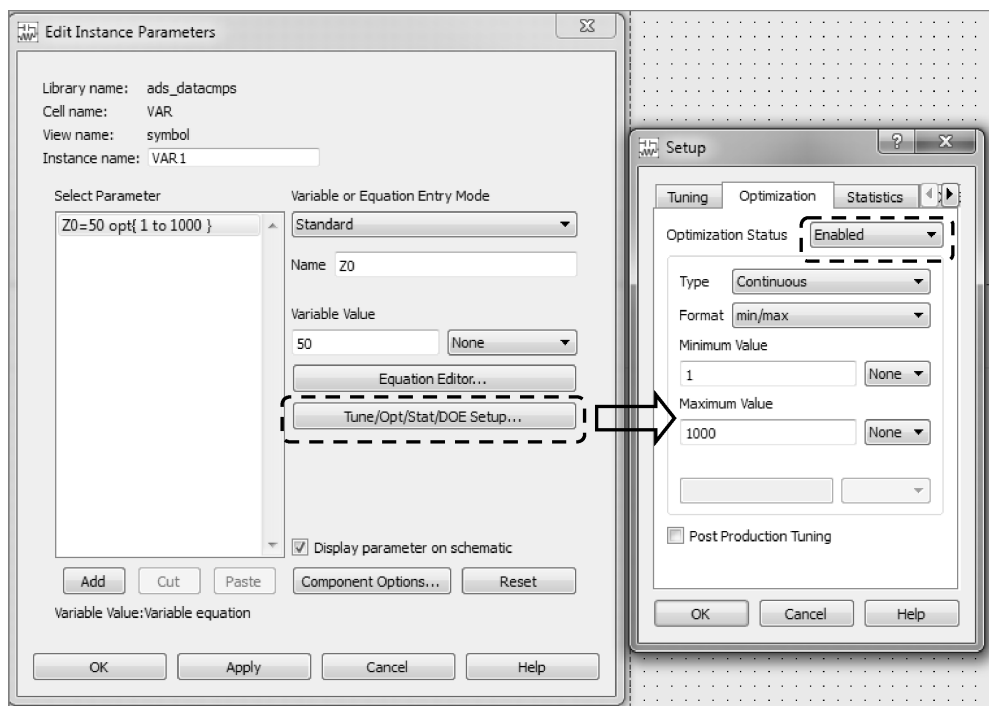


图 1-8 变量设定窗口

(5) 双击 Optim 控件，弹出如图 1-9 所示的优化设定窗口，该窗口包含 Setup、Parameters、Display 三个标签页。

☺ “Setup” 标签页：“Optimization Type” 选项用于设定优化的方式，ADS 提供了多种优化方式，最为常用的优化方式主要有随机优化、梯度优化、离散优化。随机优化支持连续或者离散的变量，梯度优化仅支持连续的变量，而离散优化只支持离散的变量。对于本范例，由于之前设定的优化变量 Z0 为连续的变量，因此选择随机优化或者梯度优化均可。对于一些优化变量和优化目标较多的电路，推荐使用先运行随机优化，再运行梯度优化的办法。

☺ “OptGoal” 选项：用户可以自己选择需要优化的目标，如果需要优化电路中所有的目标，勾选 “Use All Goals in Design”。

☺ “OptVar” 选项：用户可以指定优化的变量，如果需要电路中所有的优化变量都参与

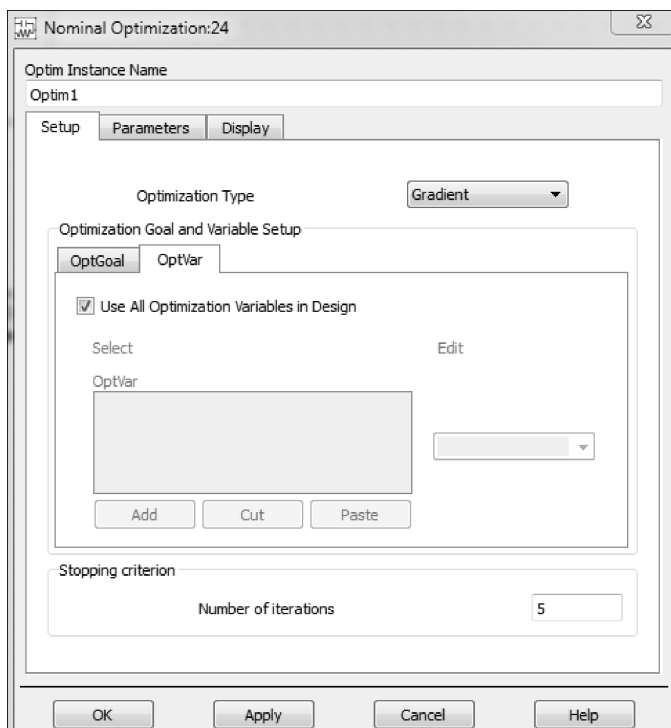


图 1-9 优化设定窗口

优化，则勾选“Use All Optimization Variables in Design”。

- ⑤ “Stopping criterion”选项：用于指定优化迭代次数，优化迭代次数的初始值为 25。对于随机优化，推荐初始值为 25 ~ 100，对于梯度优化，推荐初始值小于 10。

本例中，Setup 标签页中选择优化方式为梯度，使用电路中所有的优化目标和优化变量，优化迭代次数设为 5。对于 Optim 控件中的“Parameters”和“Display”标签页，使用默认的设置即可。

(6) 优化控件设定好以后，需要对优化目标进行设定。双击优化目标控件，在弹出的窗口的“Goal Information”标签页中有如下选项：

- ⑤ Expression：用于定义优化变量的表达式，ADS 允许优化变量为变量或者变量构成的运算公式，本例中需要优化的目标为输入或输出端口的反射系数，因此设定为“dB (S11)”。
- ⑤ Analysis：用于定义优化相关的仿真器，在本例中优化变量 S11 是需要由 S 参数仿真器“SP1”生成的。
- ⑤ Weight：用于定义优化目标的权重值，如果在电路优化中有多个优化目标，则可以根据优化变量的重要性定义目标在优化中的权重，该权重值越大表明优化变量的权重值越高。本例中由于仅包含一个优化目标，因此权重值使用默认的 1。
- ⑤ Sweep variables：该选项可以用来定义优化目标的有效频率或时域范围，用户可以设定在某个频段或时间段内对优化目标进行优化。勾选相应的频率或者时间项以后，就可以在限定线的设定选项中设置频率或者时间范围了。
- ⑤ Limit lines：该选项下可以定义优化目标的优化条件，有大于、小于、等于、区间内、

区间外总共五种定义方法，用户需要根据自己优化电路的决定选择限制范围。在本例中，优化的目标为 S11 尽可能的小，所以可设定限制线为小于 -100。代表希望的反射系数能小于 -100dB。设置好的优化目标控件如图 1-10 所示。

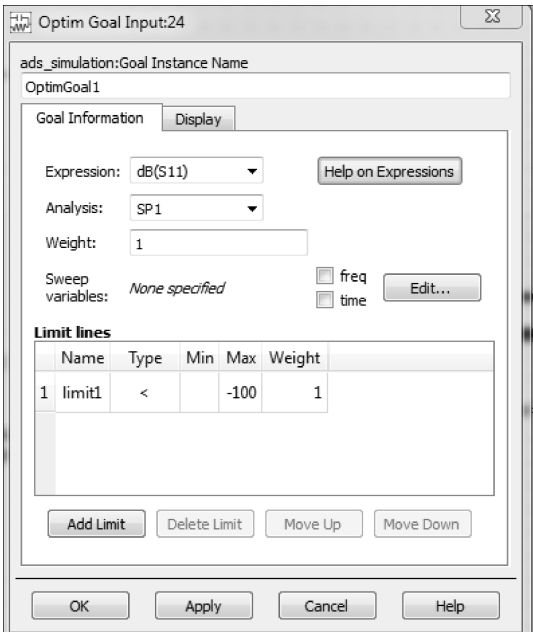


图 1-10 优化目标窗口

(7) 完成的电路如图 1-11 所示。

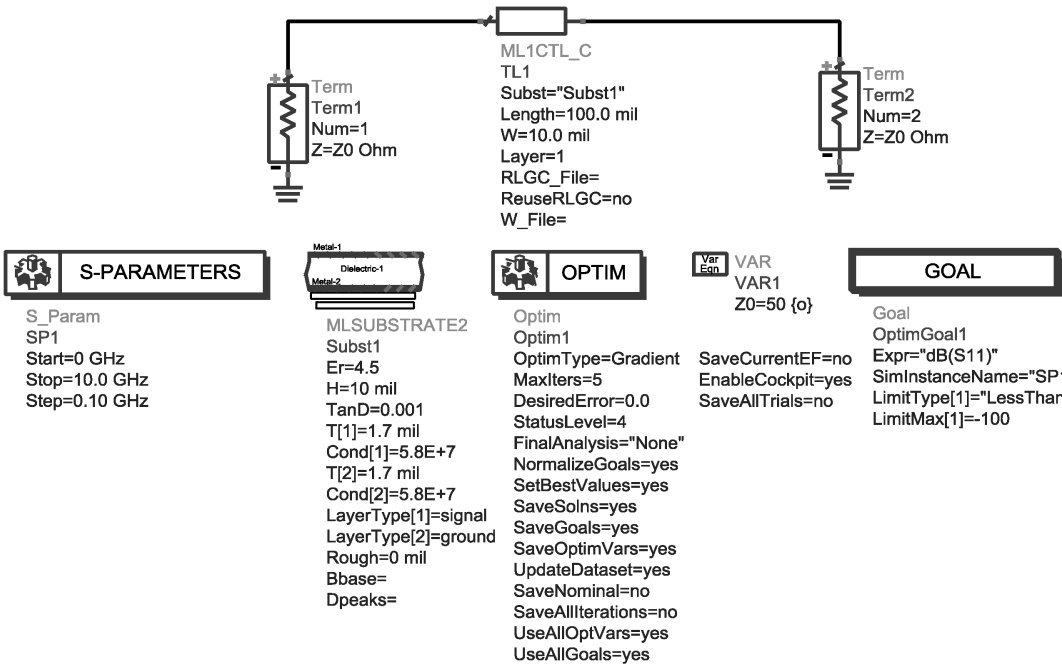


图 1-11 最终电路



## 1.3 电路优化

(1) 从 ADS2009U1 版本开始, 电路优化有单独的运行快捷方式, 在电路界面单击如图 1-12 所示的优化按钮, 电路开始优化, 同时 ADS 自动打开如图 1-13 所示的优化管理界面。



图 1-12 运行优化按钮

(2) 优化管理界面是 ADS2009U1 版本开始加入的一个新的功能, 在这个界面中, 用户可以方便地对当前的优化过程进行监控和管理, 如图 1-13 所示。

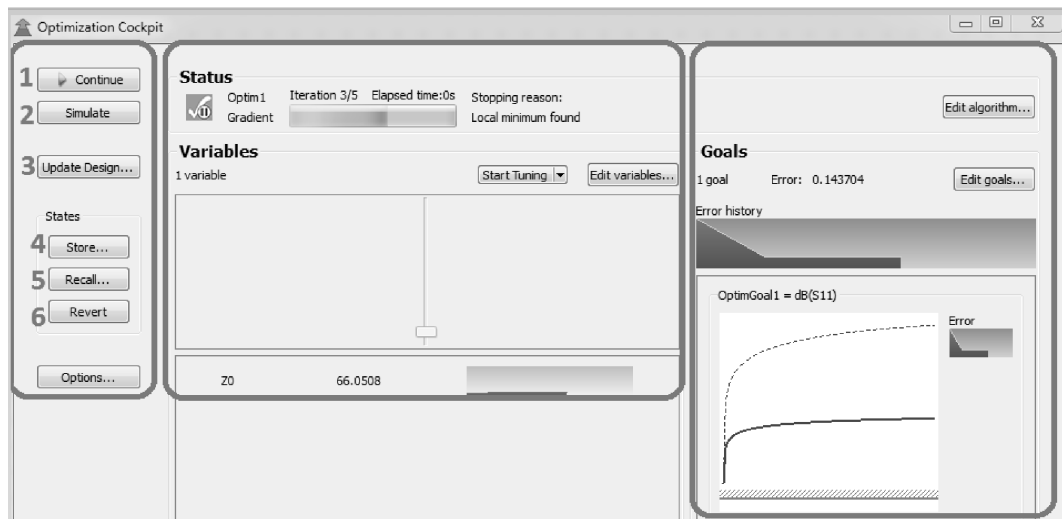


图 1-13 优化管理界面

- ☺ 最左边一列按钮, 从上到下 1~6 的按钮分别实现的功能为: 暂停/继续优化, 仿真, 更新电路, 存储当前状态, 调回已存储状态, 回到初始状态。Options 按钮可以提供一些坐标轴的选项, 大部分情况下使用默认设定即可。
- ☺ 中间的部分为优化状态栏, 可以显示优化迭代的进程以及优化变量变化的实时状态, 此处嵌入了调谐的功能, 可通过 Start Tuning 来调节变量, 从而得到变量在变化时优化目标的变化趋势。Edit variables 按钮可以调节优化变量的范围等值。
- ☺ 最右边第三部分, 为优化目标的状态栏。在该栏中, 可以通过 Edit algorithm 来更改优化方式, 也可以通过 Edit goals 选项来更改优化目标的设定。Error history 显示的是在优化的过程中误差的变化。在 ADS 的优化中, 默认设置误差为 0, 即误差越接近 0, 电路的优化就越接近目标, 假如有多个优化目标, 该窗口处还会显示每个目标对

优化的贡献度。在 Error history 栏目下面，显示的是每个目标随着优化迭代的具体变化，图中会显示三条线，一条红色的限制线，一条蓝色虚线，一条蓝色实线。红色的限制线代表的是优化目标中设定的限制线，蓝色的虚线代表优化目标的初始值，蓝色实线则代表当前迭代下的优化目标的实时值。

(3) 本例中，运行优化后经过 3 次迭代，误差已经为 0.143 左右。由于反射系数不可能达到理想的没有任何反射的情况，因此优化到这个误差时已经可以认为端口阻抗值和传输线的特征阻抗是一致的，在结果文件窗口中得到的特征阻抗为 66.051 Ohm，如图 1-14 所示。

OPTIM.Z0	
	66.051

图 1-14 电路优化结果

(4) 得到结果后，关闭优化窗口，此时会弹出窗口询问是否要将优化变量的结果更新到电路，选择“否”即可。



## 1.4 版图仿真设定

前面介绍了在原理图中使用原理图微带线模型的方法，该方法中使用的传输线是基于数学解析方法的模型，在 ADS 中也可以使用电磁模型得到更准确的结果。

(1) 将原理图“Single\_Line”使用“Layout→Generate/Update layout”功能进行同步，将会弹出对话框，在此使用默认的设定，然后单击 OK 按钮。

(2) ADS 会自动生成版图，如图 1-15 所示，单击保存。

(3) 如图 1-16 所示，单击工具栏中的 EM setup 按钮，EM setup 的设定是 ADS 版图仿真最关键的步骤，刚打开的 EM setup 界面左边会显示有三个警示，这些警示表示这个版图仿真缺少层叠、端口等的设定。下面将重点介绍层叠的设定和 EM-model 生成的过程，对于 EM setup 菜单中其余的端口、网格等设定，请参考实例 10 内容。

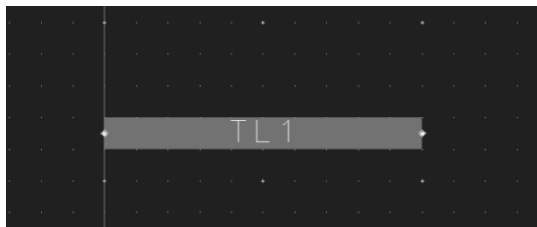


图 1-15 单端传输线版图

(4) 在 Substrate 子菜单中，单击 New。在弹出的窗口中直接使用默认选项，然后单击 Apply。在随后弹出的层叠设定窗口中，单击 File→import→Substrate from Schematic，选择原理图“Single\_Line”，导入的层叠保存为“Single\_Line.subst”，如图 1-17 所示。

(5) 双击打开 Single\_Line.subst，如图 1-18 所示，可以看到层叠包含两层金属层，分别是名为 cond 的上层信号层和底层的 cover 层，cover 层即理想的封闭地平面。两层金属之间夹了一层 Subst\_1 的介质材料。左键单击浅蓝色的介质层，在窗口右侧可以看到这一层介质的属性。

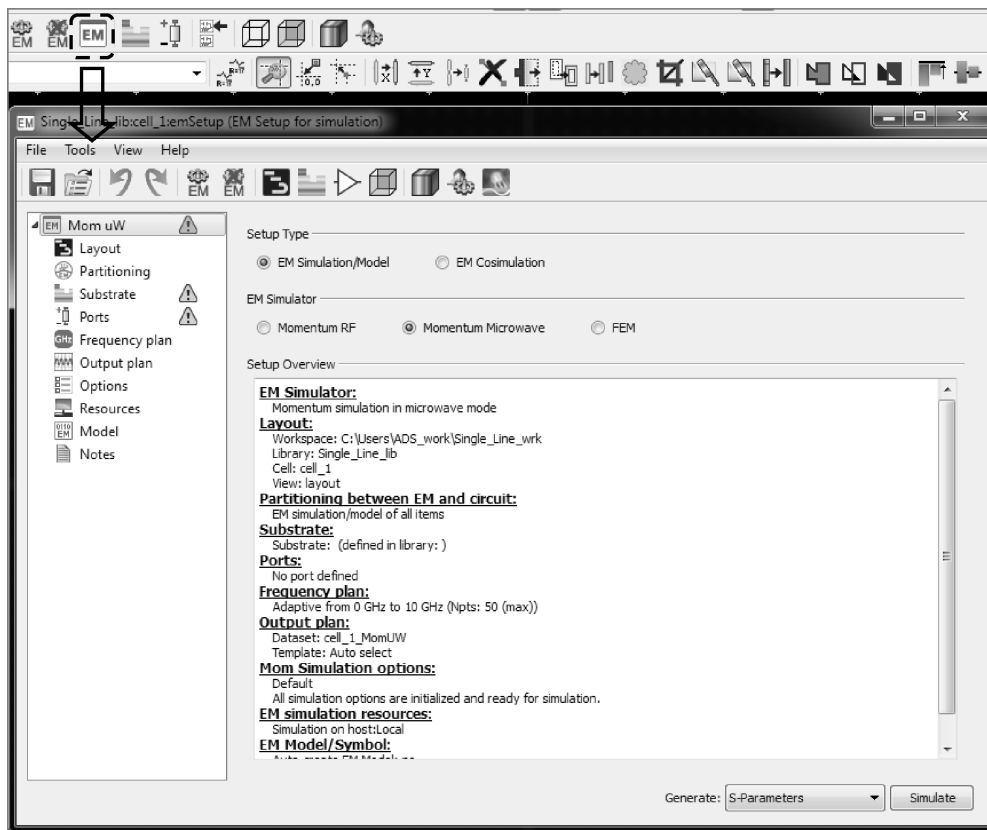


图 1-16 EM setup 窗口

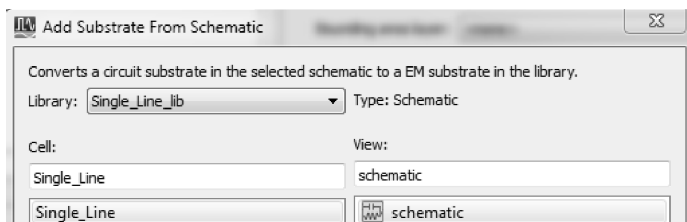


图 1-17 从电路导入层叠设定

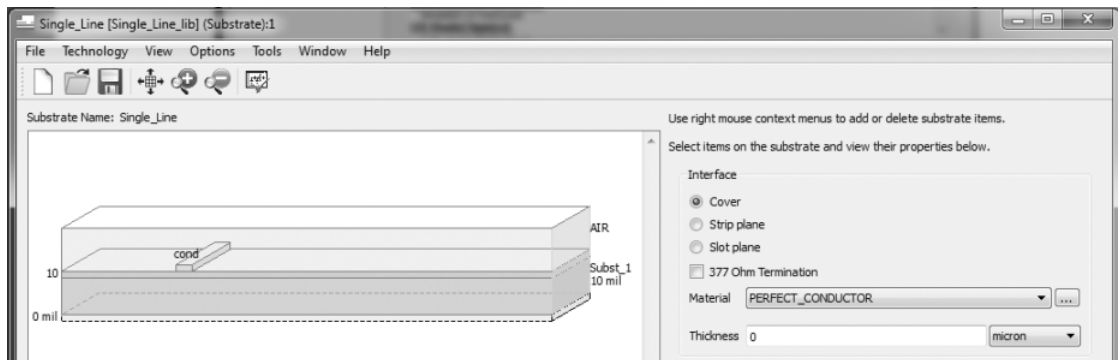


图 1-18 层叠属性窗口

(6) 单击材料 Subst\_1 旁边的浏览 (...) 按钮可以查看或编辑介质的属性信息。如图 1-19 所示, 在弹出的材料编辑窗口中, 分别添加三个不同介电常数的介质 Dielectric\_1、Dielectric\_2、Dielectric\_3, 介电常数分别为 4、4.5 和 5。

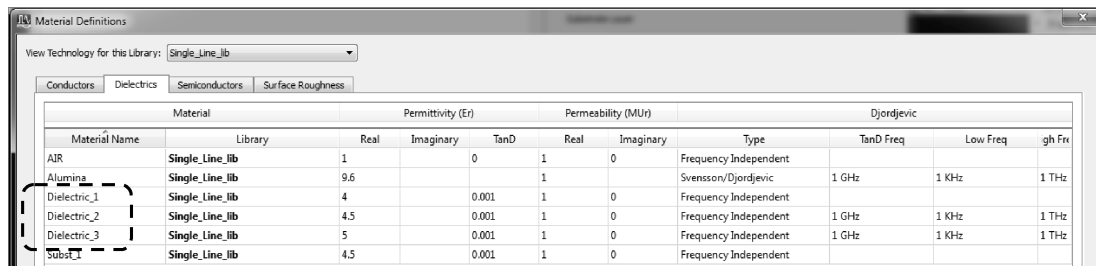


图 1-19 材料设定窗口

(7) 将 Single\_Line.subst 使用 File→Save a Copy As 另存三次, 分别命名为 “Single\_Line\_Dk1”、“Single\_Line\_Dk2”、“Single\_Line\_Dk3”。完成上述操作后, 在 ADS 主界面会显示新增的三个层叠定义, 如图 1-20 所示。如果没有出现, 请关闭该 workspace 再重新打开。

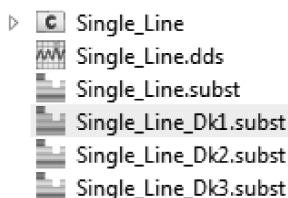


图 1-20 多个层叠定义

(8) 打开 Single\_Line\_Dk1.subst, 左键单击浅蓝色介质, 窗口右侧出现介质的属性信息, 在材料的下拉菜单中选择 “Dielectric\_1”, 单击保存。同样的, 分别将 Single\_Line\_Dk2.subst 和 Single\_Line\_Dk3.subst 中的介质材料更改为 “Dielectric\_2” 和 “Dielectric\_3”, 然后保存。

(9) 在 Layout 中, 分别在传输线两端按照如图 1-21 所示的方法添加端口。

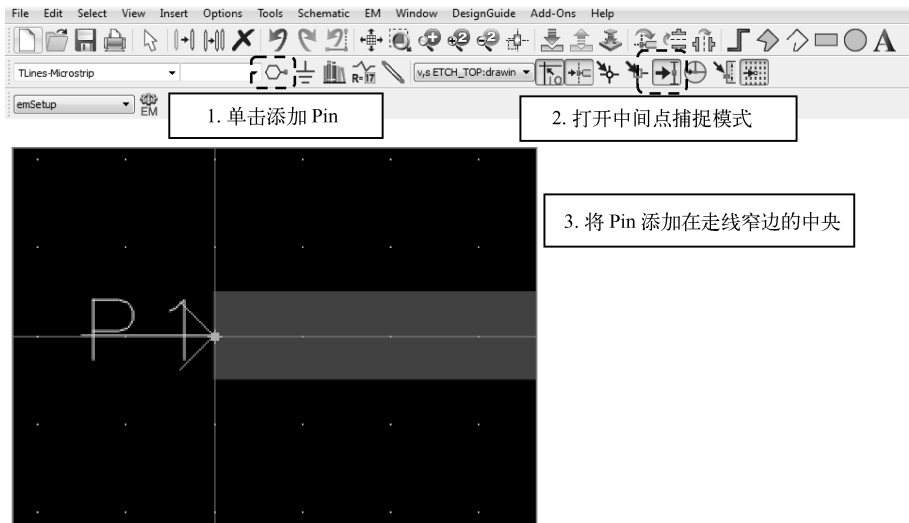


图 1-21 添加端口

(10) 单击 EM Setup 按钮, 依次确认如下选项的设置:

☺ Substrate 选项的下拉菜单中选择层叠为 “Single\_Line\_Dk2”。

☺ Frequency Plan 选项下设置仿真频段为 0GHz 到 10GHz, 扫描方式为 Adaptive。

⑤ 在 Options 选项的 Mesh 标签下，勾选 Edge Mesh（边缘网格选项）。

上述设置完成后，在确认 EM Setup 窗口没有黄色警示出现后，单击保存。

(11) 回到 Layout 界面，单击 File→Design Parameters，其中 Cell Parameters 标签页的设置如图 1-22 所示，此处为这个版图设计添加了一个字符串参数“Substrate”，该字符串的初始定义为“Single\_Line\_Dk2”，除了该默认值外，此版图设计的层叠“Substrate”还有“Single\_Line\_Dk1”和“Single\_Line\_Dk3”两种其他设定，在稍后进行的共仿真中可以通过扫描该字符串的值，来仿真不同介电常数的介质下传输线特征阻抗。

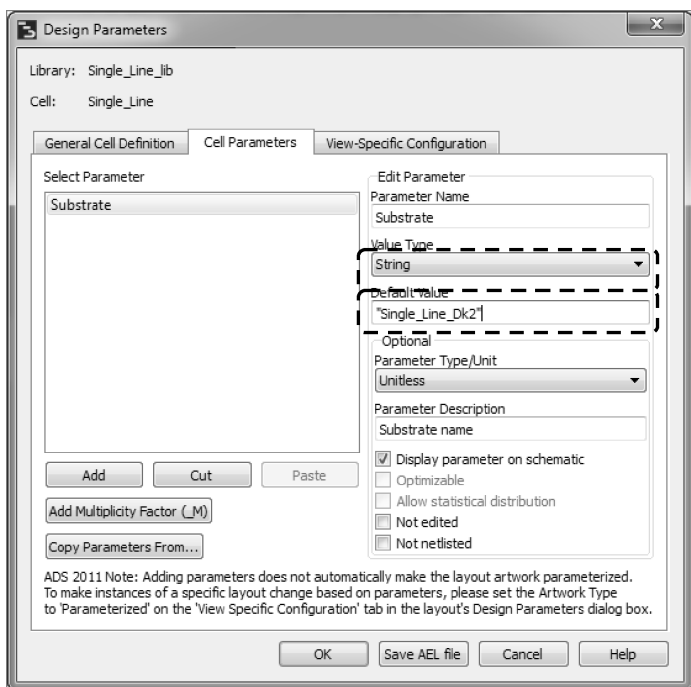


图 1-22 设计参数定义



## 1.5 电磁仿真

(1) 完成上述的版图仿真设定步骤以后，在版图界面单击 EM 菜单，选择 Component→Create EM Model and Symbol，如图 1-23 所示。选择创建 EM Model 和符号以后会弹出对话框确认，单击 OK 按钮即可。完成后会提示创建成功，这样就将刚才的传输线版图模型创建成了一个可以在原理图中调用的元件。

(2) 打开之前的电路“Single\_Line”，单击 File→Save a Copy As，另存为一个新的设计单元下的电路“single\_line\_MOM\_Dk\_sweep\_OPT”，如图 1-24 所示。

(3) 在“Component history”栏中输入“Single\_Line”，如图 1-25 所示。这样就可以在该电路中将刚生成的版图元件调用，用该版图元件替换传输线的电路模型“ML1 CTL\_C”。



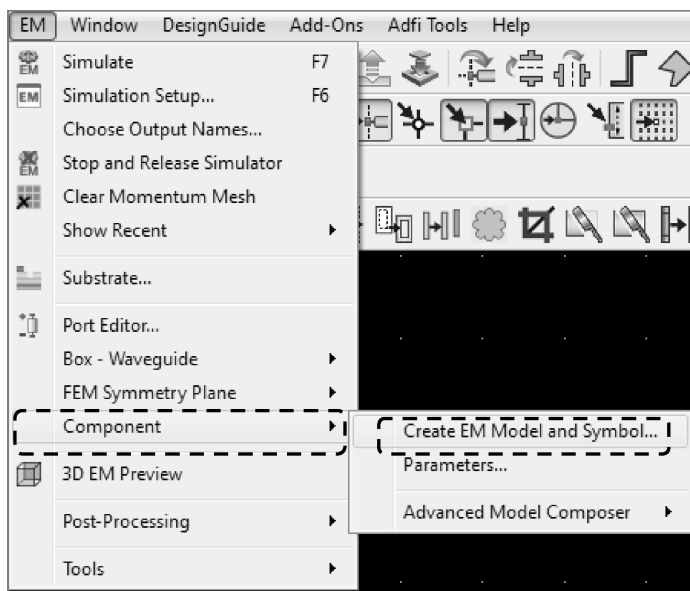


图 1-23 创建版图元件

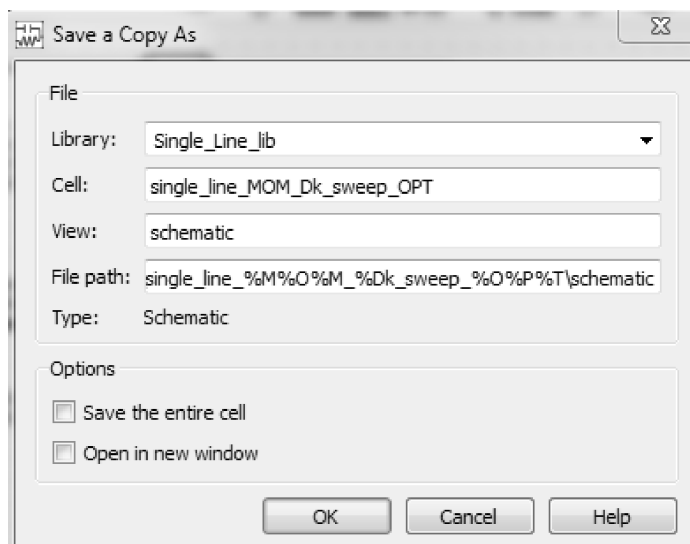


图 1-24 新建设计单元

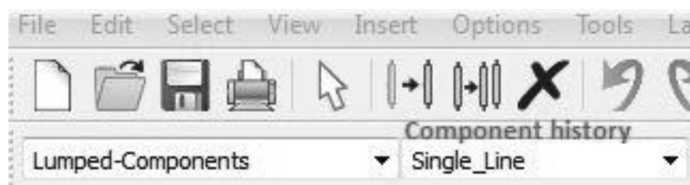


图 1-25 在电路中调用版图元件

按照图 1-26 所示搭建原理图。

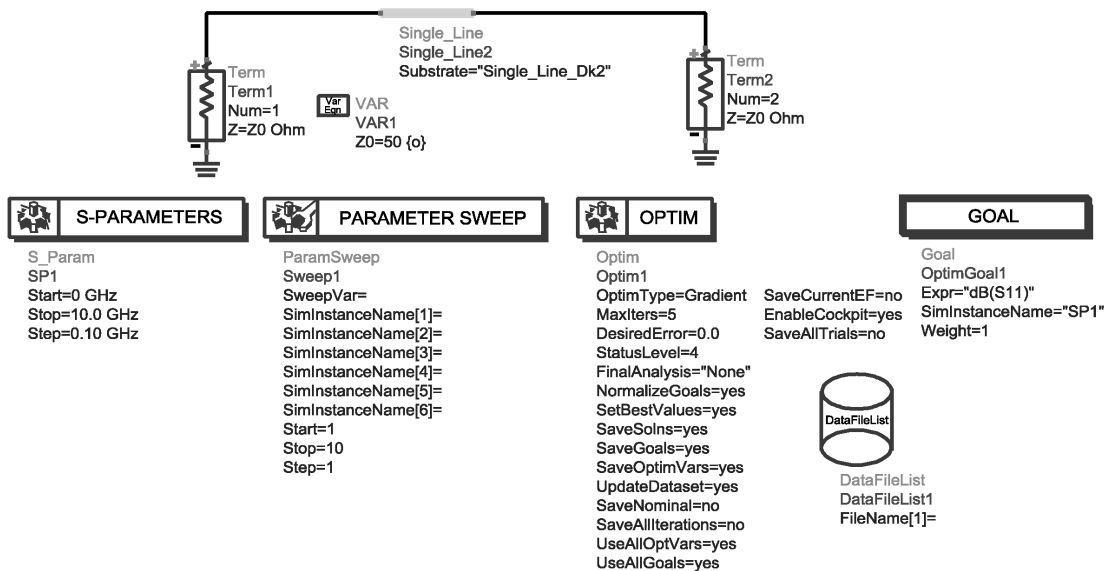


图 1-26 电磁仿真初始电路

(4) 如图 1-27 所示, 在 DataFileList 元件中, 分别将三种层叠定义文件读入, 并且将 Index 项的值设置为 “ndx”。注意在输入层叠文件名的时候, 需要在层叠文件名前加上 “Single\_Line\_lib:”, 该前缀的目的在于申明层叠文件来自于哪个库, 本例中只有一个和项目 Single\_Line\_wrk 同名的库, 因此使用该库名即可。将 Index 定义为变量 ndx, 这样在稍后扫描三种层叠文件时扫描变量 ndx 即可, 而无须直接扫描层叠文件的名称。

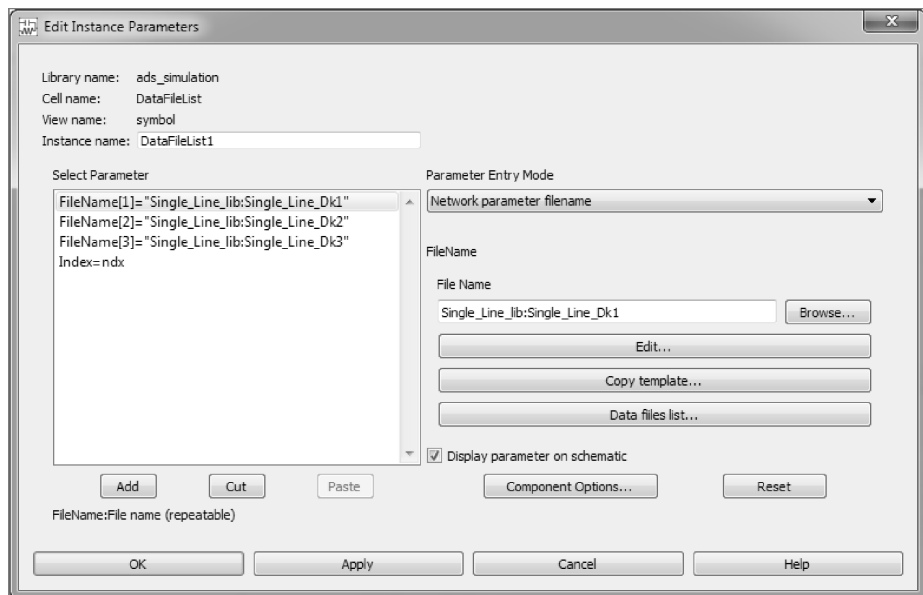


图 1-27 DataFileList 元件设定

(5) 添加两个变量分别为 “substrateName” 和 “ndx”, 按照图 1-28 所示设置两个变量的初始值分别为 “DataFileList1. FileName” 和 “1”。

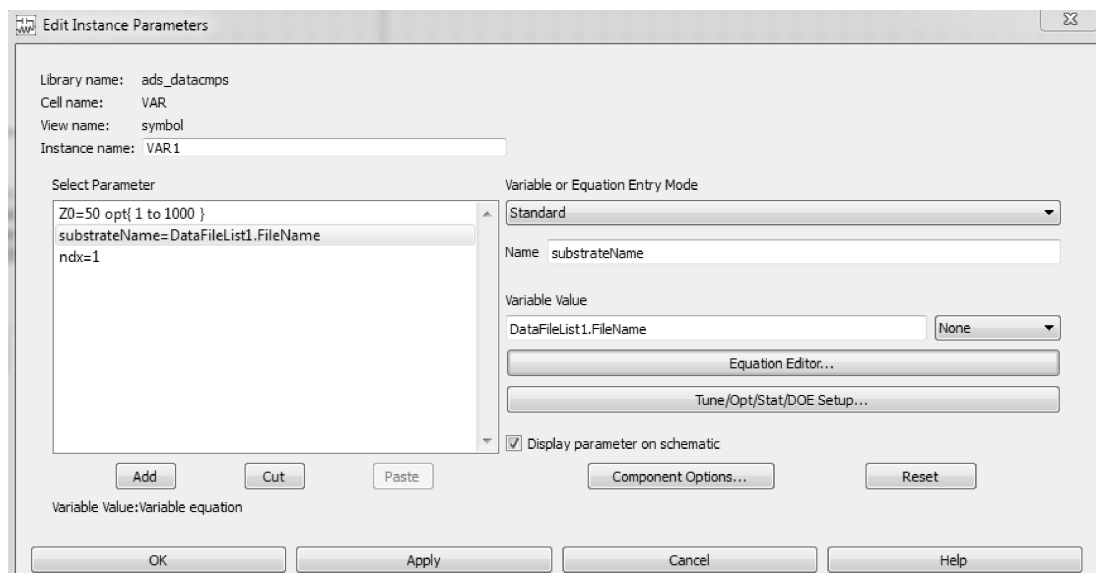


图 1-28 变量设定

(6) 双击 Parameter Sweep 控件，按照图 1-29 所示进行参数设定。

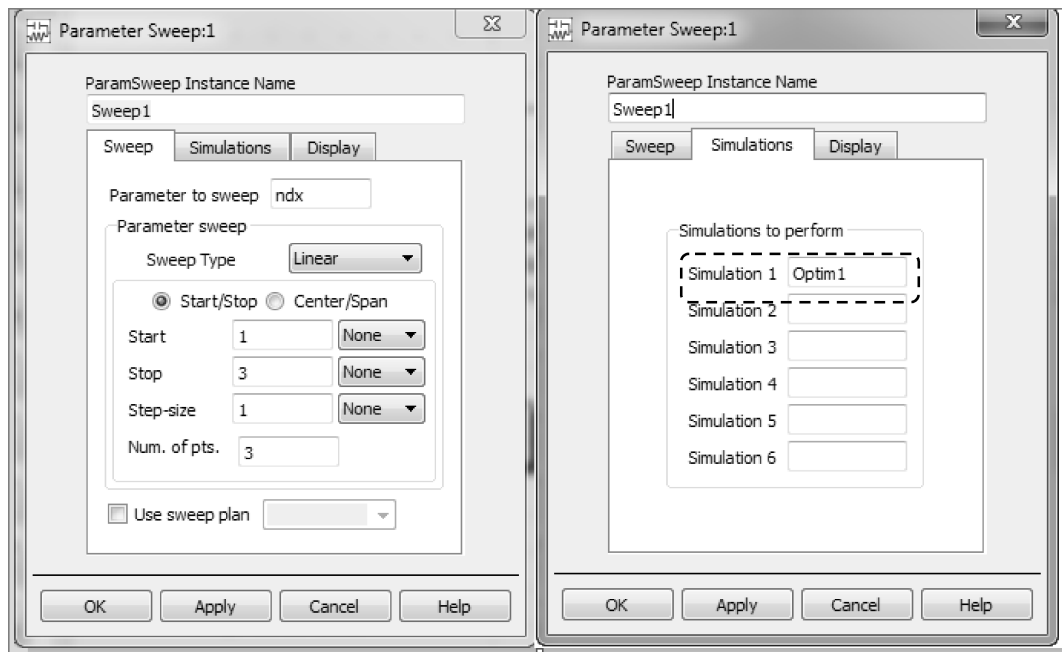


图 1-29 参数扫描设定

(7) 最终的电路如图 1-30 所示。

(8) 单击优化按钮，电路开始优化仿真。在弹出的优化管理界面中可以看到进行了三次扫描优化，如图 1-31 所示。

(9) 在弹出的结果窗口中添加如图 1-32 所示的数据列表结果，并通过单击 Insert→Text 菜单来添加注释来说明扫描优化得到的特征阻抗结果分别对应的介电常数。

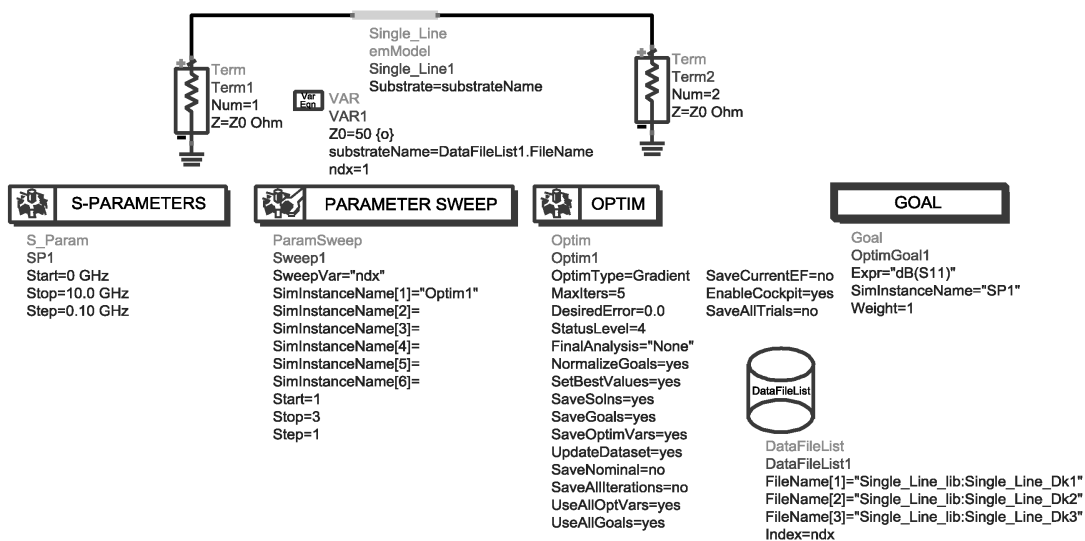


图 1-30 电磁仿真最终电路

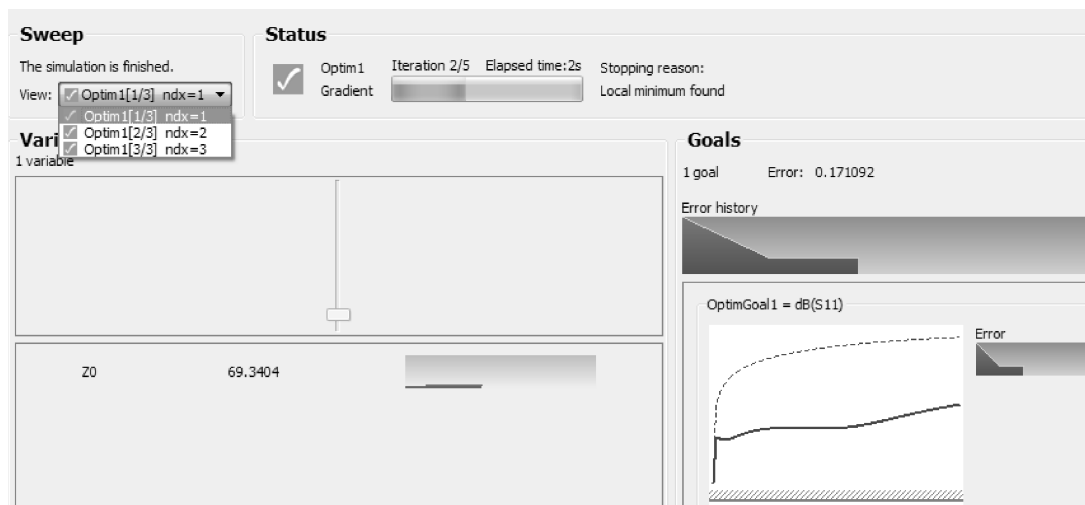


图 1-31 共仿真优化进程

OPTIM.Z0	
69.340	Er=4.0
65.988	Er=4.5
63.087	Er=5.0

图 1-32 共仿真结果

(10) 在本书实例下载资源中的工程文件“Single\_Line\_wrk”中还有一个设计单元“single\_line\_MOM\_thickness\_sweep\_OPT”，该设计单元同样为共仿真扫描层叠文件求解传输线特

征阻抗，只是层叠文件中变化量为介质的厚度。该单元实验步骤与扫描介电常数的设计单元很相似，这里不再赘述，读者可根据实例下载资源附带工程文件尝试完成。



## 1.6 CILD 工具使用

在前面的内容中，介绍了如何使用电磁仿真的办法求得不同介质下的单端传输线特征阻抗，但是这种方法稍微有些烦琐，那么有没有比较方便快捷的计算工具呢？

从 ADS2014.01 版本开始，添加了一个新的工具 Controlled Impedance Line Designer，简称为 CILD。这个工具可以方便地读取 ADS 版图仿真中定义的层叠文件，然后选取模型（微带线、带状线），输入线的线宽、长度等值以后，可以计算阻抗、反射系数、时延等指标，并且可以对层叠的参数和传输线尺寸进行扫描和统计分析。

下面将介绍如何使用 CILD 工具来分析之前设计单元中的单端传输线。

(1) 打开 Single\_Line 设计单元下的版图，在版图界面选择 Tools→Controlled Impedance Line Designer，如图 1-33 所示。

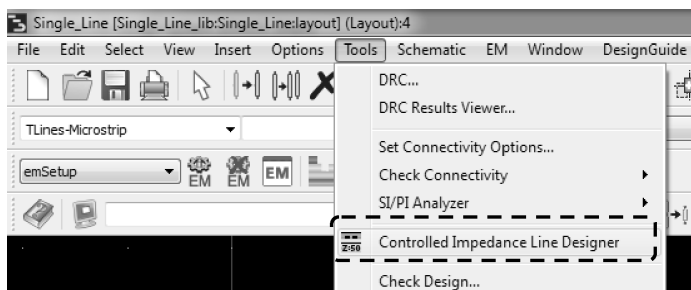


图 1-33 CILD 工具开启

(2) 在 CLID 界面中，左侧为层叠设置和模型选取界面，右侧上半部分为参数定义界面，下半部分为分析结果显示界面。按照图 1-34 所示，做如下设定：

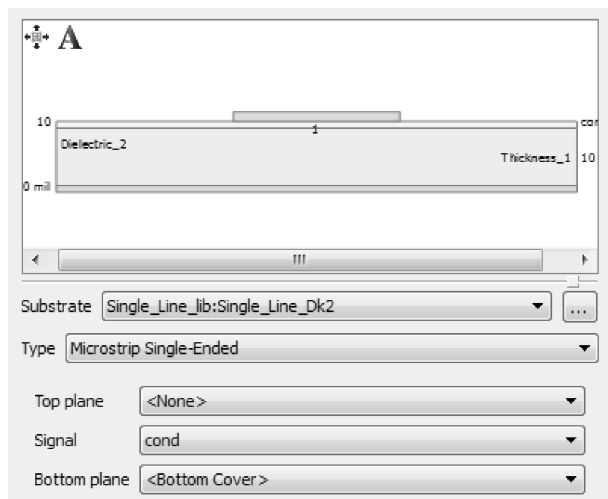


图 1-34 CILD 工具层叠设定

⑤ Substrate 下拉菜单中，选择层叠文件 “Single\_Line\_lib: Single\_Line\_Dk2”。

⑥ Type 下拉菜单中选择 “Microstrip Single\_Ended”。即读取层叠文件 Single\_Line\_Dk2，选取单端微带线模型。读入层叠以后，层叠的上平面、信号层、底平面的定义分别为 “None”、“Cond” 和 “Bottom Cover”。

(3) 在右侧的参数设定栏目中选择 “Analyze”，输入传输线长度为 100mil，宽度为 10mil，频率使用默认的 1GHz 即可，如图 1-35 所示。

(4) 按下如图 1-36 所示的运行按钮。

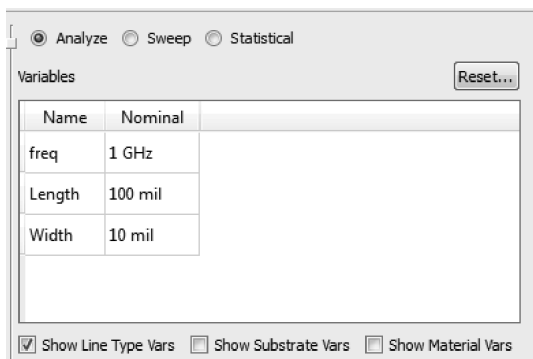


图 1-35 CILD 工具传输线参数设定



图 1-36 CILD 工具运行

(5) 运行片刻后，右侧的结果栏目中显示出关于传输线的各项参数，如图 1-37 所示。可以看到求解得到的特征阻抗为 66.1672Ohm。

Electrical		
	Real	Imag
Zc (ohm)	66.1672	-0.260355
Gamma (1/m)	0.17649	36.8828
Attenuation (dB/mil)	3.89375e-05	0
Attenuation (dB)	0.00389375	0
Delay (ns/mil)	0.0001491	0
Delay (ns)	0.01491	0
Propagation Velocity (m/s)	1.70355e+08	0
Effective Dielectric Constant	3.09692	0

图 1-37 CILD 工具结果窗口

(6) 在右侧的参数设定栏目中选择 “Sweep”，勾选 “Show Material Vars” 选项，此时层叠介质的材料属性将显示在窗口中，勾选 “Dielectric\_2\_Er\_Real” 为可扫描的变量，扫描范围为 4 到 5，步进为 0.5，如图 1-38 所示。

(7) 单击运行后，在弹出的结果显示窗口中，可以看到如图 1-39 所示的特征阻抗曲线。

(8) 为了方便数据对比，在此我们将其转化为数据列表显示。双击该结果图，在弹出的窗口中选择数据列表形式，如图 1-40 所示。

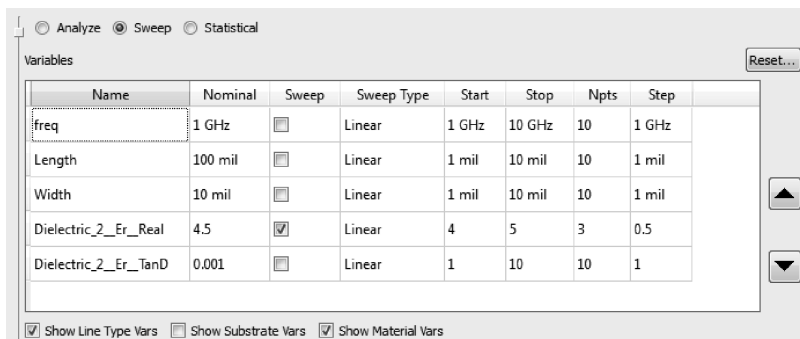


图 1-38 CILD 扫描传输线参数

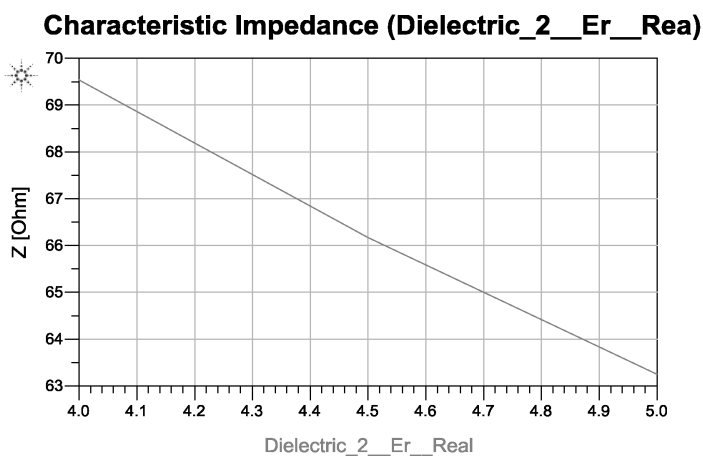


图 1-39 CILD 工具扫描介电常数结果

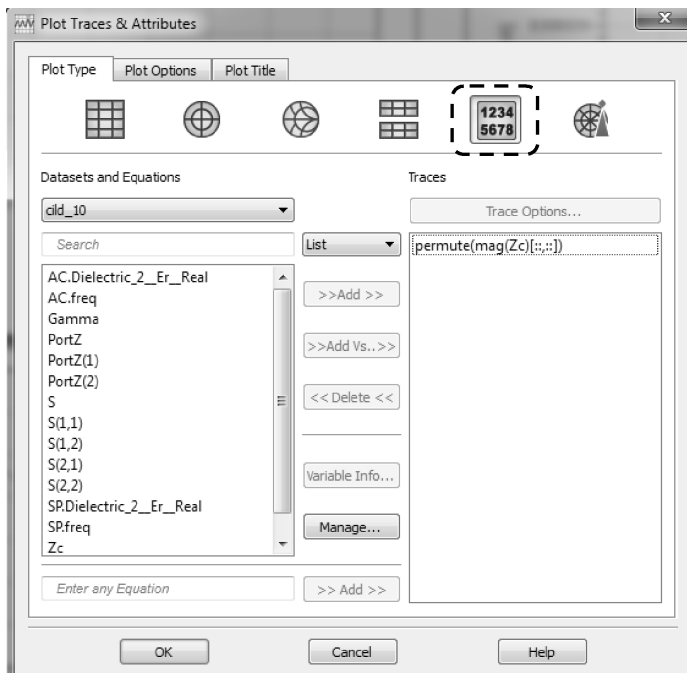


图 1-40 数据列表格式转换

(9) 得到的结果如图 1-41 所示, 从结果中可以看到介质介电常数为 4、4.5、5 时, 求得传输线特征阻抗分别为 69.538 Ohm、66.168 Ohm 和 63.249 Ohm。保存该结果显示窗口并且关闭。

**Characteristic Impedance (Dielectric\_2\_\_Er\_\_Real)**

Dielectric_2__Er__Real	permute(mag(Zc){:,:,:}) freq=1.000 GHz	
4.000		69.538
4.500		66.168
5.000		63.249

图 1-41 数据列表结果

(10) 对于扫描不同介质厚度的情况, 只需要勾选 “Show Substrate Vars”, 然后将 “Thickness” 选为扫描变量, 设置扫描范围运行即可, 读者可自行尝试。

### 【总结】

通过上述三种单端传输线的特征阻抗分析方法, 得到表 1-1 所示的结果。在介电常数为 4.5 时, 三种方法的结果非常接近。在扫描介电常数的情况下, 电磁仿真和 CILD 的结果误差也是很小的。在仿真时间方面, CILD 是最快捷的, 电磁仿真方法则需要相对较长的时间。

**表 1-1 三种方法结果对比**

介电常数 Zc 求解法	4.0	4.5	5.0
电路求解		66.051 Ohm	
共仿真	69.340 Ohm	65.988 Ohm	63.087 Ohm
CILD	69.538 Ohm	66.168 Ohm	63.249 Ohm

对于单端传输线阻抗分析, 应该根据所分析的微带线模型来选择适当的方法。对微带线、带状线结构, 可以选择 Linecalc 工具或者优化的方法。对于层叠结构为多层的复杂结构, 应当首选 CILD 工具。对于导入 CAD 版图中的传输线, 可以采用电磁仿真的方法求解特征阻抗。



## 实例 2 差分传输线分析

### 【目的】

本实例主要介绍差分传输线的分析方法，包括差分线的串扰、奇偶模阻抗的求解及差模共模阻抗的求解。另外，单端/差分 S 参数的转换及 SNP 文件导入也是本章将要涉及的内容。



### 2.1 创建 ADS 项目文件

打开 ADS 主程序，新建一个工程项目，命名为 Diff\_Line\_Analysis\_wrk，如图 2-1 所示。新建项目时，Technology 选择 “Standard ADS Layers, 0.0001 millimeter layout resolution”。

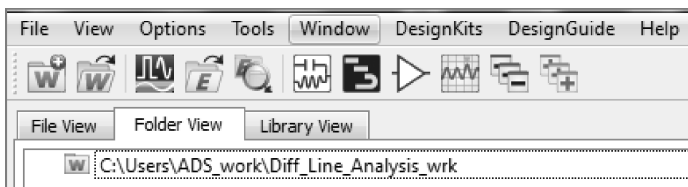


图 2-1 新建工程项目



### 2.2 差分线版图元件创建

(1) 在建好的项目 Diff\_Line\_Analysis\_wrk 中，新建版图 “EM\_Board\_Diff\_Trace1”，如图 2-2 所示。

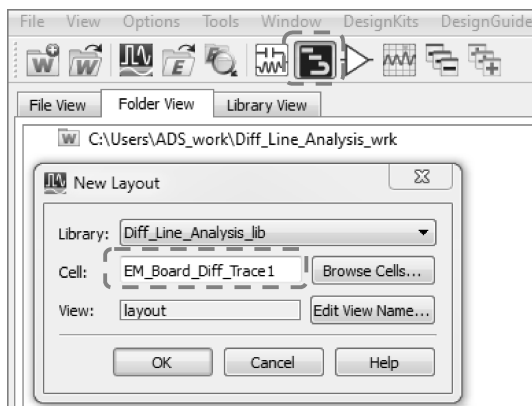


图 2-2 新建版图窗口

(2) 从“TLines\_Multilayer”元件库中选择“ML2CTL\_C”元件放入版图界面。在“ML2CTL\_C”四端加上 Pin 脚，得到如图 2-3 所示的版图效果。“ML2CTL\_C”的参数为：Length = 50mm，W = 0.9495mm，S = 0.5mm，其余项目保持默认设定即可。



图 2-3 版图效果

(3) 单击 EM Setup 按钮，在 EM Setup 界面中，选择仿真器为 EM Simulation/Model 选项下面的 Momentum Microwave。

(4) 在 EM Setup 界面单击新建 Substrate，新建如图 2-4 所示的层叠结构。

☺ 介质材料的介电常数为 2.94。

☺ 损耗角正切为 0.0012。

☺ 介质厚度为 0.381mm。

☺ 金属材料为铜。

将新建的层叠命名为 z\_Board\_sub。

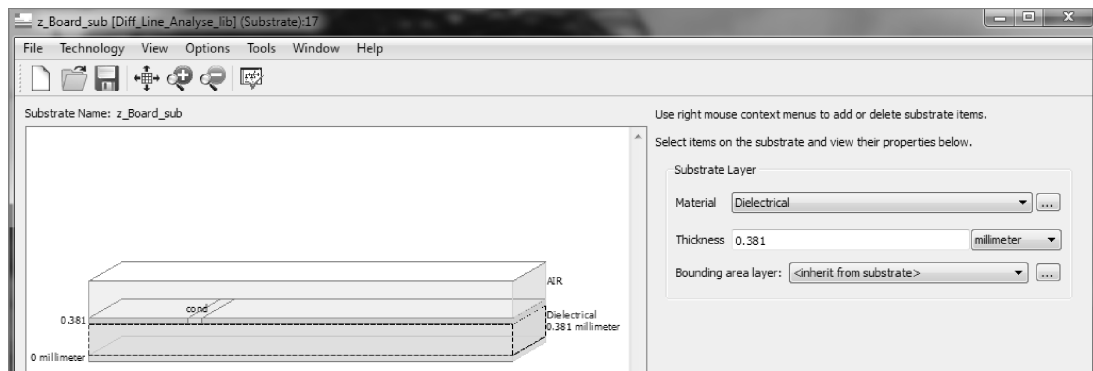


图 2-4 新建层叠

(5) 回到 EM Setup 窗口，确认警告已消失。在 ADS 版图环境中，选择 EM→Component→Create EM Model and Symbol，选择创建 EM Model，单击 OK 按钮。



## 2.3 差分线损耗串扰仿真

(1) 新建原理图名为 cross\_talk。选择 Source - Time Domain 元件库中的 VtSine，使用该正弦信号激励差分线。

(2) 按照图 2-5 所示建立电路。

(3) 在建好的电路中添加节点 v1、v2、v3、v4、vsrc。

(4) 在电路中加入瞬态仿真器和 S 参数仿真器。添加如下变量：

☺ Freq = 10GHz。

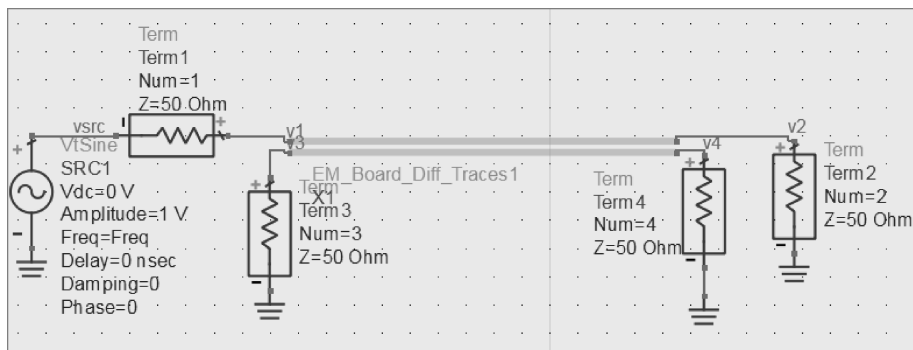


图 2-5 添加电路节点

- ☺  $T_{\text{freq}} = 1/\text{Freq}$ 。
- ☺  $T_{\text{step}} = T_{\text{freq}}/20$ 。
- ☺  $T_{\text{start}} = 0 * T_{\text{freq}}$ 。
- ☺  $T_{\text{stop}} = T_{\text{start}} + 10 * T_{\text{freq}}$ 。

S 参数仿真器设定如下：

- ☺ 起始频率为 0.1GHz。
- ☺ 截止频率为 10GHz。
- ☺ 频率步进为 50MHz。

最终电路如图 2-6 所示。

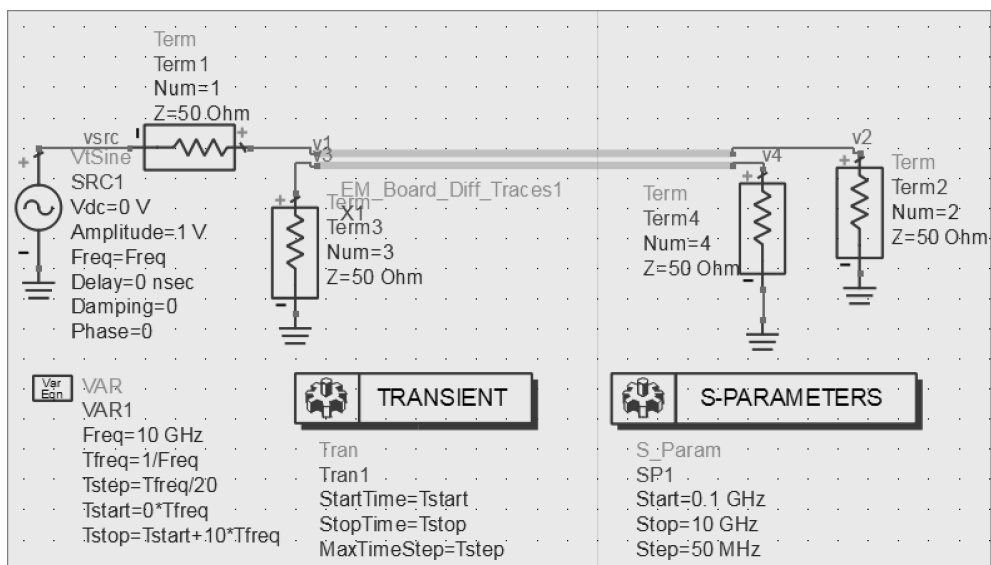


图 2-6 最终电路

(5) 运行仿真，在结果中显示如图 2-7 所示的电压曲线。从该电压曲线图中可以看到在各个节点处电压在时域中的变化情况，vsrc 为激励源电压，v1 为激励源电压和反射电压的叠加，v2 则为输出端电压，可以看到一段明显的时间延时，这是由于信号经过差分线所引起的，v3 为近端串扰，v4 为远端串扰。

(6) S 参数的结果 S11、S21、S31 和 S41 如图 2-8 所示。

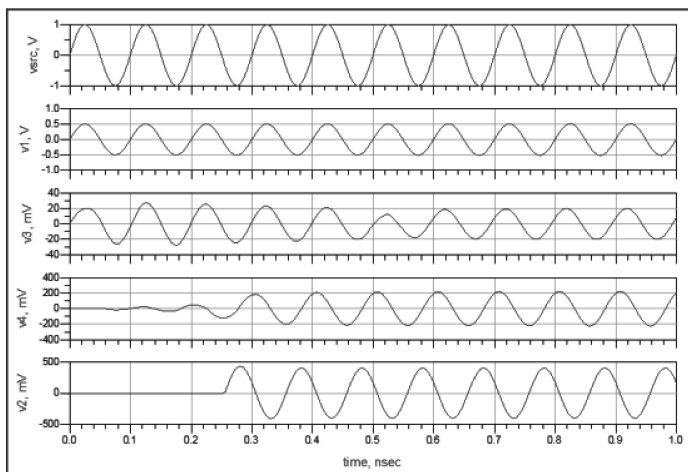


图 2-7 电压曲线

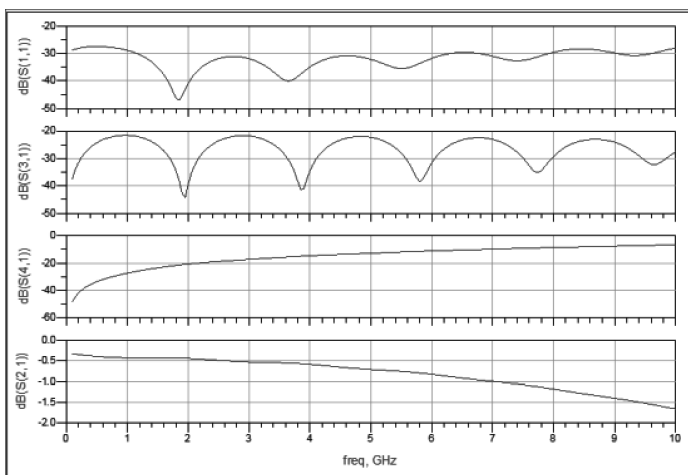


图 2-8 S 参数结果



## 2.4 差分线阻抗分析

差分线由于结构的特殊性，根据激励方式的不同，需要关注其奇/偶模阻抗及共模/差分阻抗。在此介绍优化方法、CILD 工具以及混合模定义算法。

(1) 建立如图 2-9 所示的原理图 diff\_imp\_OPT。将 Zdiff 设置为优化变量，设定优化目标、优化控件，可以得到如图 2-10 所示的结果，Zdiff 为 92.210Ω。

(2) 参考 1.6 节，启动 CILD 工具，基板选择 z\_Board\_sub，线型选择 Microstrip Edge - Coupled，按照如图 2-11 所示设定，得到的 Zdiff 为 91.980Ω，Zcom 为 27.670Ω。

(3) 新建原理图，命名为 diff\_imp\_THEORY。从 Sources - Freq Domain 元件库中选择交流电流源 I\_AC，按照如图 2-12 所示搭建电路。

(4) 在同一个原理图中，再使用电流源用共模的激励方式接入差分线，如图 2-13 所示。

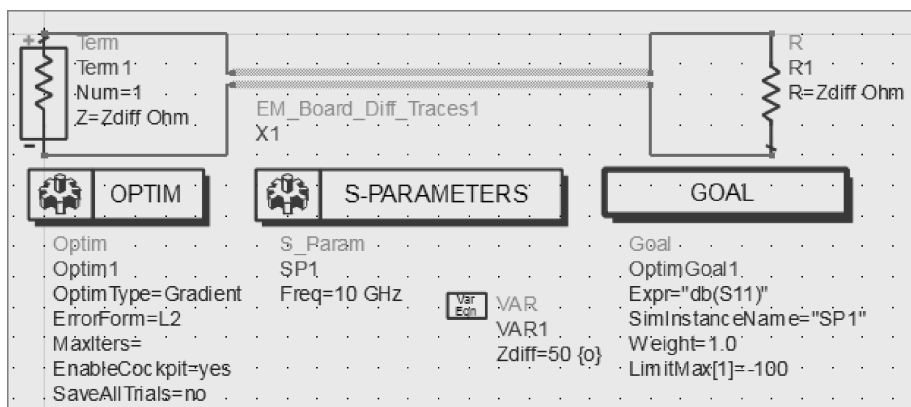


图 2-9 优化方法电路

Zdiff
92.207

图 2-10 优化方法 Zdiff 结果

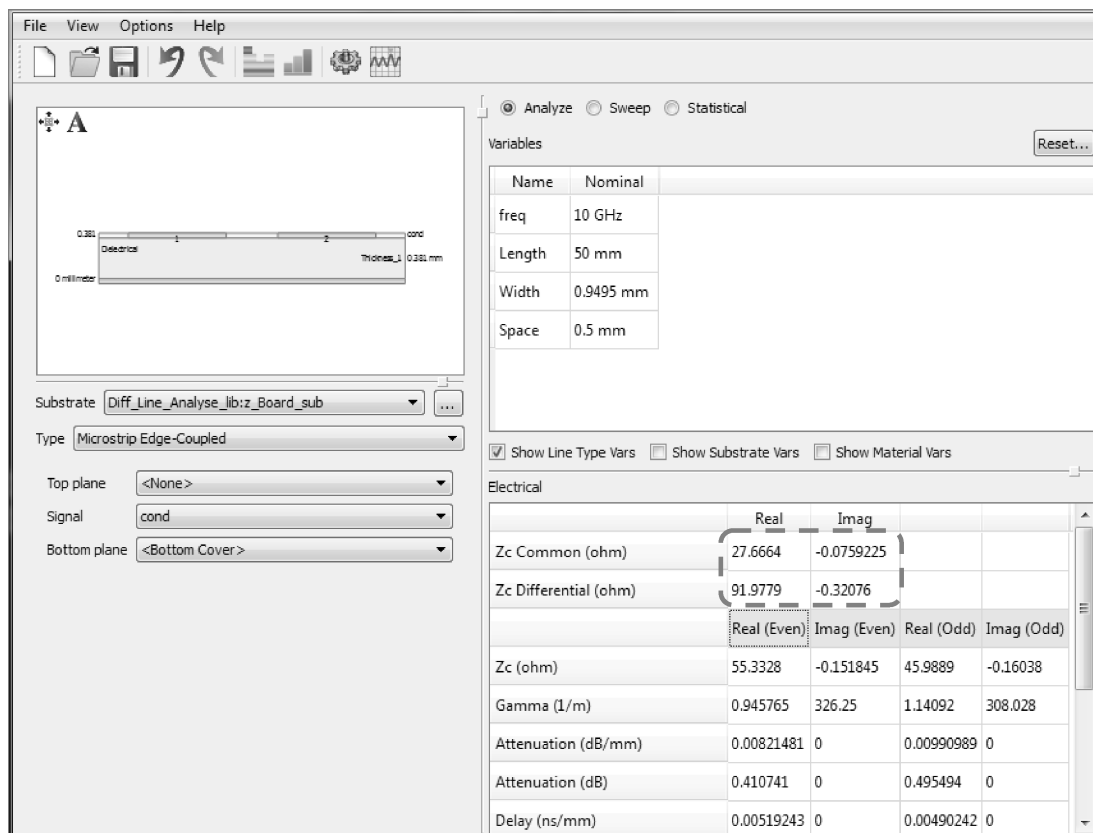


图 2-11 CILD 计算结果

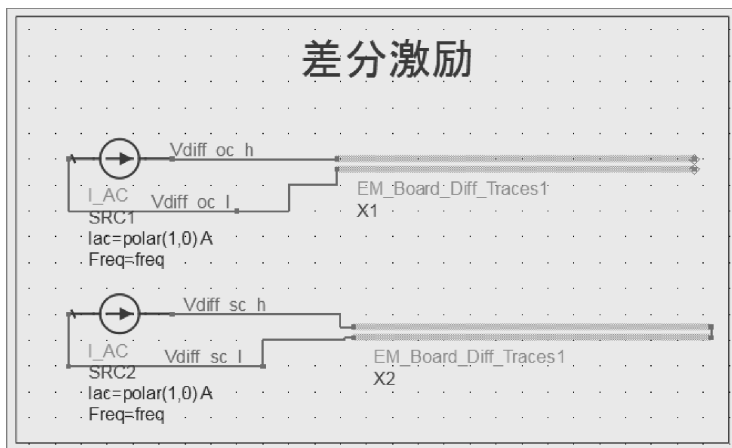


图 2-12 差分激励连接

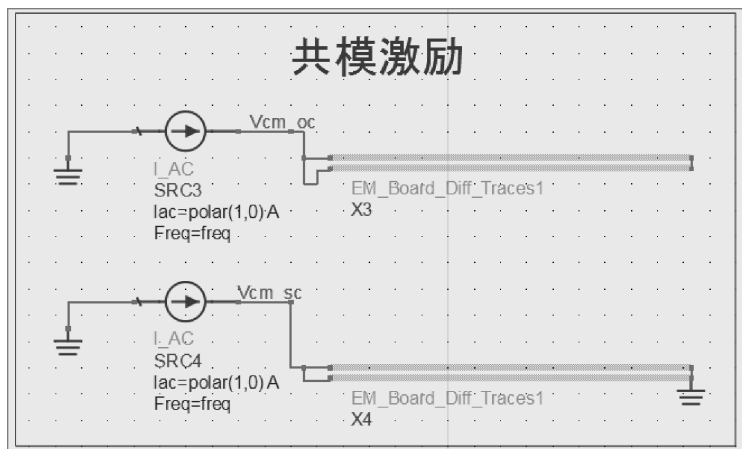


图 2-13 共模激励连接

(5) 从 Simulation - AC 元件库中添加 AC 仿真器，并且设置仿真频率为 10GHz。从相同的元件库中加入 MeasEqn，双击 MeasEqn，添加如图 2-14 所示的公式。这些公式为差分/共模阻抗以及奇/偶模阻抗的推导公式。

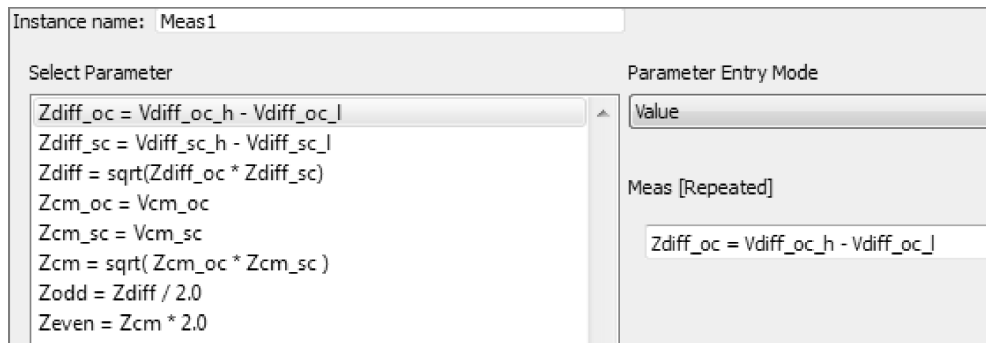


图 2-14 公式添加

(6) 最终电路如图 2-15 所示。

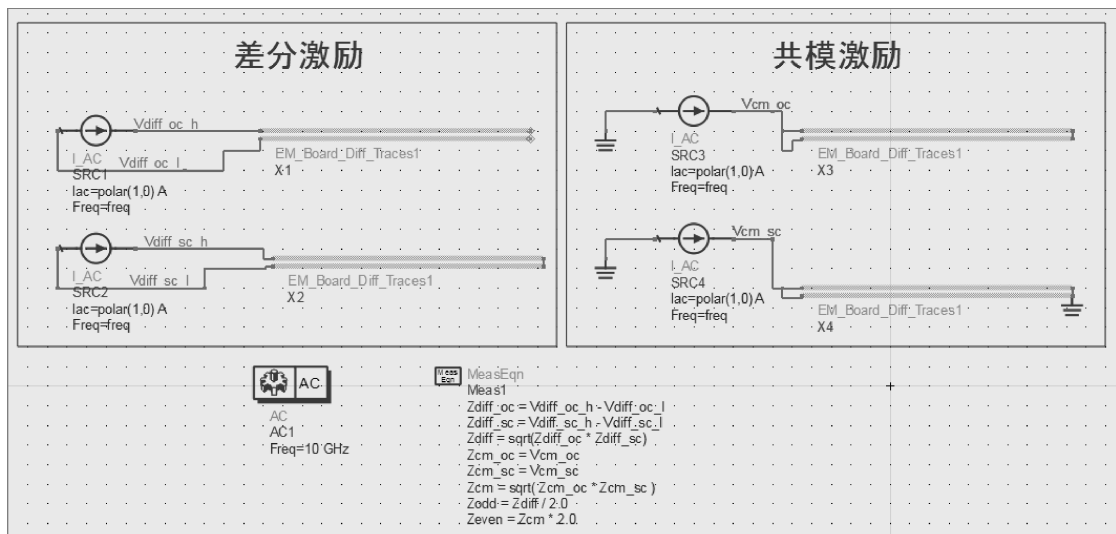


图 2-15 最终电路

(7) 运行仿真，在结果窗口中添加  $Z_{diff}$ 、 $Z_{cm}$ 、 $Z_{odd}$ 、 $Z_{even}$ ，如图 2-16 所示，通过和前两种方法的对比可见，三种方法的结果是很吻合的。

$Z_{cm}$	$Z_{diff}$	$Z_{even}$	$Z_{odd}$
27.722 / -0.370	92.207 / -0.162	55.443 / -0.370	46.103 / -0.162

图 2-16 阻抗结果



## 2.5 单端 S 参数到差分/共模 S 参数的转换

之前的实验得到了差分线的单端 S 参数，对于混合模 S 参数，可以使用 ADS Design Guide 中的模板来求得。

(1) 打开刚才建立的电路设计单元 diff\_imp\_THEORY，然后在顶部菜单中选择 Design-Guide→Signal Integrity Applications，在弹出的窗口中选择 Mixed Mode Simulation→Four Port Simulation，如图 2-17 所示，然后单击 OK 按钮。

(2) 单击 OK 按钮以后会自动生成名为 4port\_template 的原理图，只需要将中间的四端口网络替换成差分线版图元件，设置仿真频段，差分阻抗即可运行仿真，如图 2-18 所示。

(3) 在电路菜单中选择 Simulate→Simulation Settings，在弹出的窗口中勾选 Open Data Display when simulation completes，如图 2-19 所示，确保仿真完毕后就能弹出结果显示窗口。

(4) 运行仿真，弹出结果。结果由 Mixed-mode equation、Single-ended S-parameters、Mixed-mode S-parameters 和 Alternative mixed-mode formulation 四页组成。在 Mixed mode S-parameters 页面下，可以找到混合模 S 参数的结果，如图 2-20 所示。

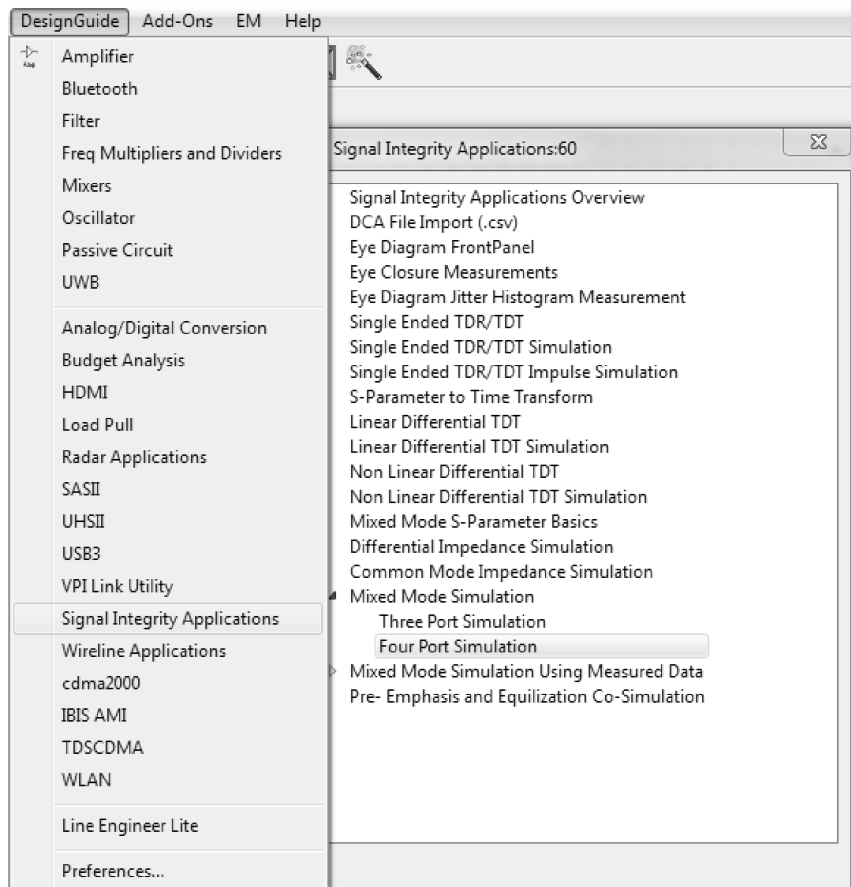


图 2-17 模板选择

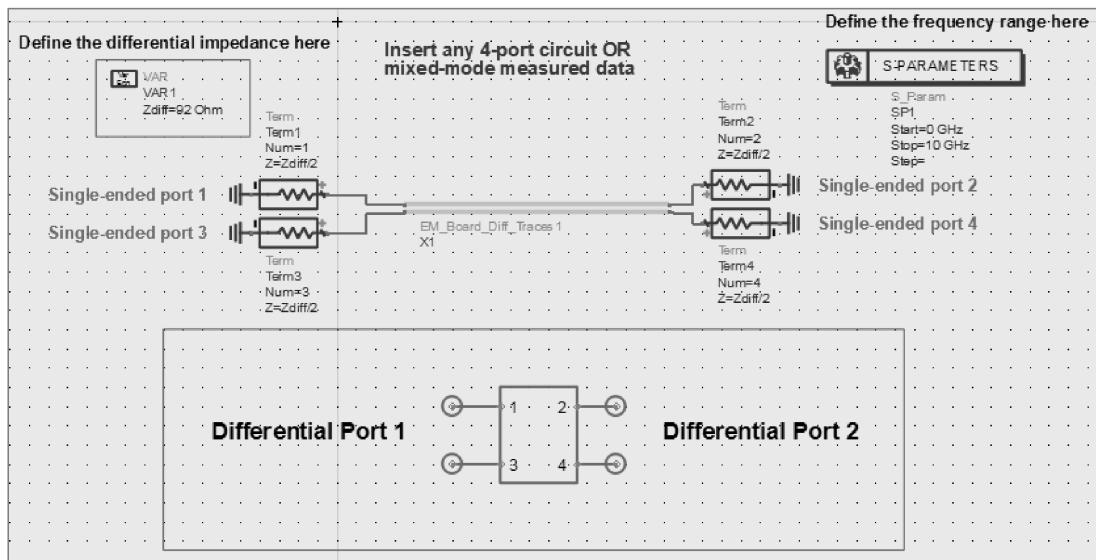


图 2-18 最终电路



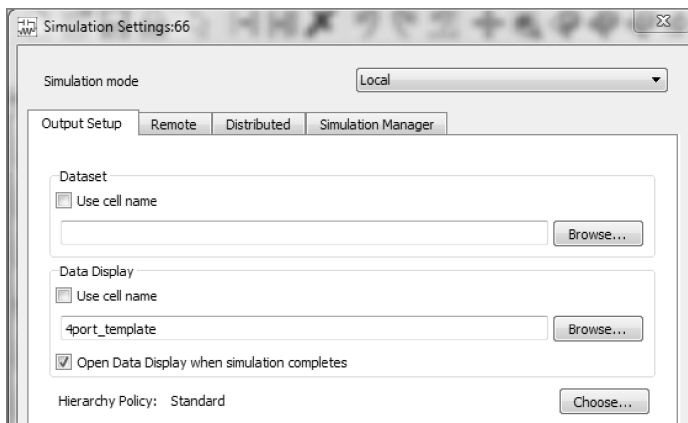


图 2-19 仿真设定

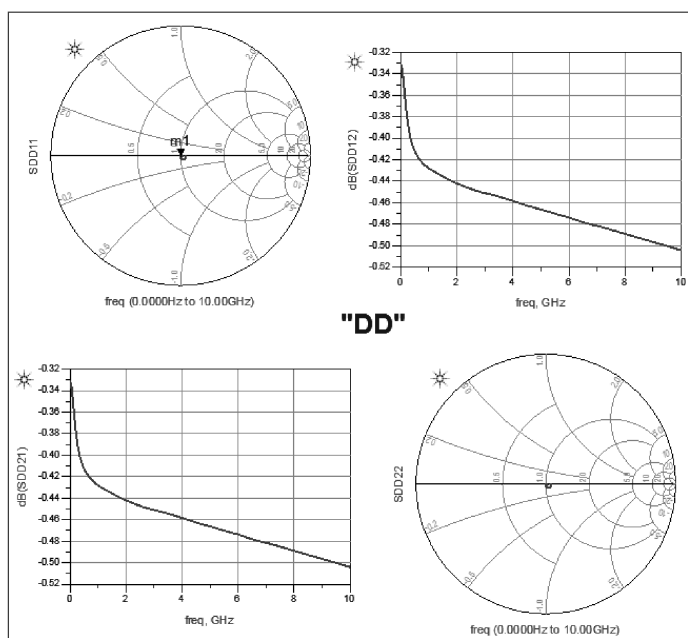


图 2-20 差分 S 参数结果



## 2.6 SnP 文件导入

通道仿真中，经常需要将测量得到的 SnP 文件导入 ADS 中进行仿真。

(1) 请从本书实例下载资源中找到 Snp\_Uilities\_Add\_On\_v1\_5 这个压缩文件，然后将其解压在电脑中 ADS 安装路径下面。

(2) 回到 ADS 主界面，在菜单中选择 Tools→Manage ADS AEL Addons，在弹出的窗口中单击“Add User ADS AEL Addon”按钮，将会弹出如图 2-21 所示的窗口。

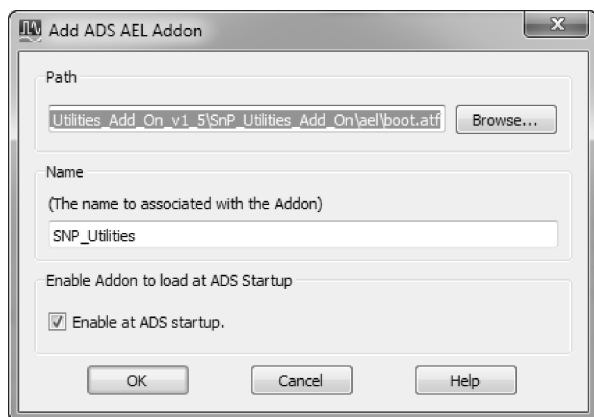


图 2-21 安装插件窗口

(3) 在路径指定项中指定压缩包下面的 SnP\_Uilities\_Add\_On\ael\boot.atf 文件，在名称项目中输入“SNP\_Uilities”，然后勾选 Enable at ADS startup，单击 OK 按钮确认。

(4) 重启 ADS，打开任意电路设计单元，电路界面的菜单栏中会出现一个新的选项“Add - Ons”，单击该选项下的插件“SnP Utilities”，如图 2-22 所示。

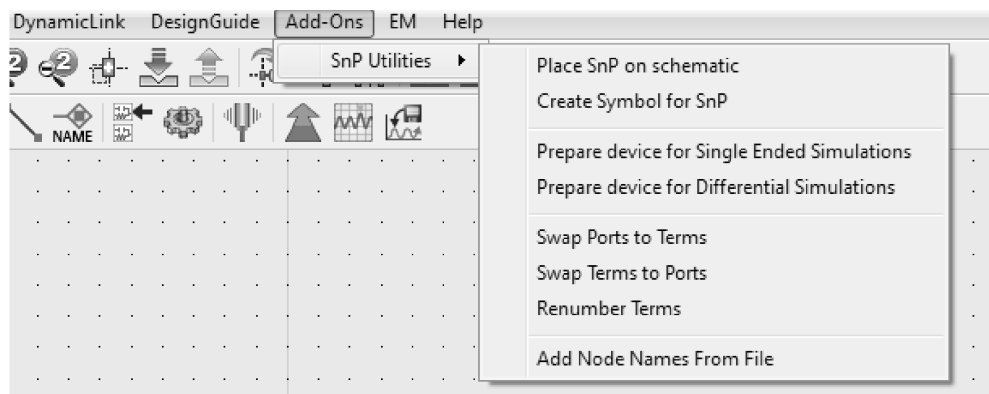


图 2-22 插件安装效果

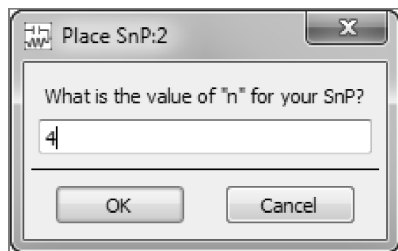


图 2-23 输入端口数目

(5) 新建原理图，命名为 import。在 SnP Utilities 选项单击 Place SnP on schematic。此时弹出窗口要求填入所导入 S 参数的端口数，输入 4，如图 2-23 所示，单击 OK 按钮确认。

(6) 在 S4P 元件中，将实例下载资源中的“C166EM.s4p”加载进去。单击“Create Symbol For SnP”，然后根据 Pin 脚排序选择 Symbol 类型，如图 2-24 所示。

(7) 新建原理图，命名为 import\_verify。在元件历史栏目中输入“import”，回车，如图 2-25 所示。

(8) 在 SnP Utilities 选项下面，单击“Prepare device for Differential Simulations”。ADS 将自动添加端口、巴伦以及 S 参数仿真器。参数设定如下：

☺ 差分阻抗 1000hm。

- ☺ 起始频率 0GHz。
- ☺ 截止频率 20GHz。
- ☺ 步进 10MHz。

最终电路如图 2-26 所示。

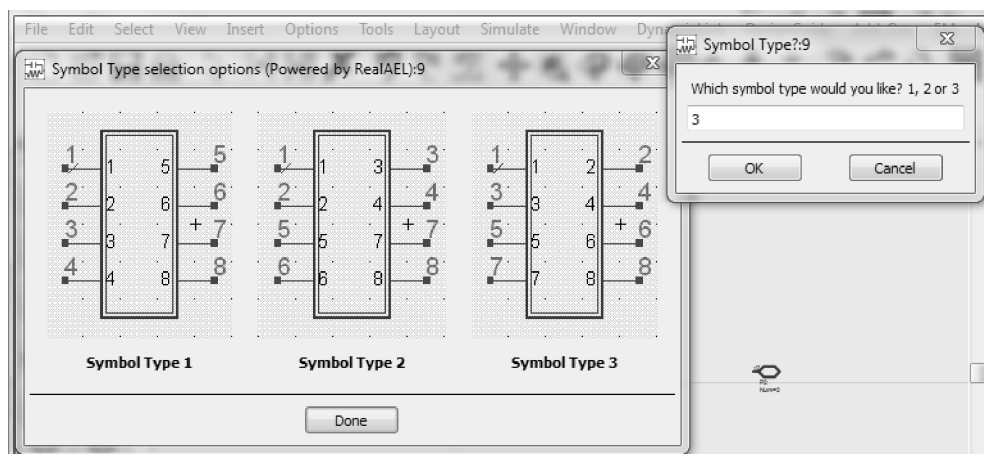


图 2-24 元件符号类型选择

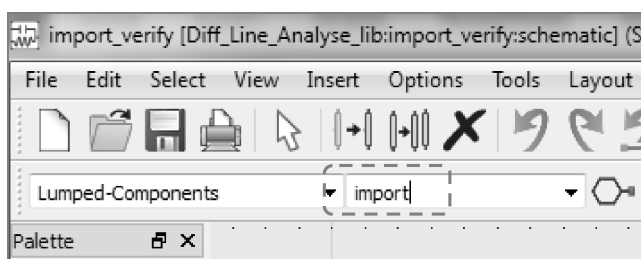


图 2-25 调用元件

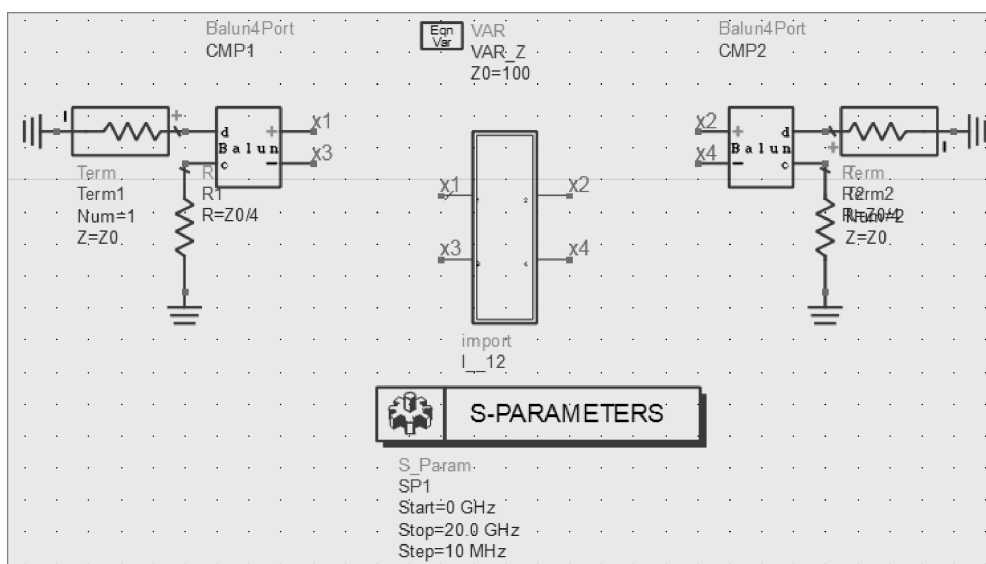


图 2-26 最终电路

(9) 运行仿真，得到的 S 参数结果如图 2-27 所示。

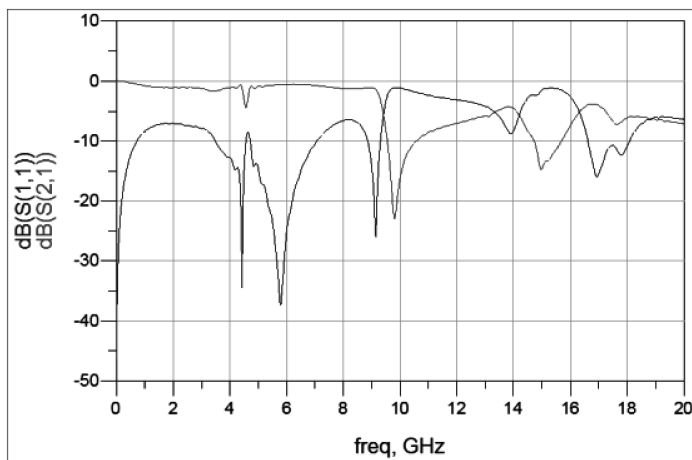


图 2-27 差分 S 参数结果

(10) 该 S4P 文件可直接使用 ADS Design Guide 的模板（参考 2.5 节）来计算差分 S 参数，如图 2-28 所示。

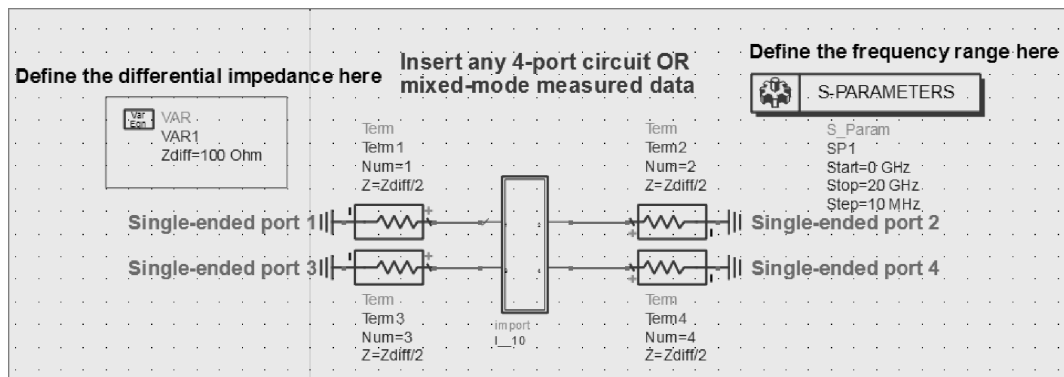


图 2-28 使用模板求解

(11) 运行仿真，差分 S 参数结果将显示在 Mixed - mode S - parameters 页面中的“DD”部分，将电路 import\_verify 得到的结果显示在同一结果窗口中，通过对比，两种方法得到的差分 S 参数结果完全吻合，如图 2-29 所示。

### 【总结】

本实例通过对差分线一些重要特性的分析，介绍了差分线阻抗和差分线 S 参数计算的几种方法。通过对比，CILD 工具在前仿阶段，可以准确地进行差分线阻抗的设计。对于 SnP 文件的导入，使用 SnP Utilities 这个小插件会提升效率，混合模理论的方法可以用于验证。

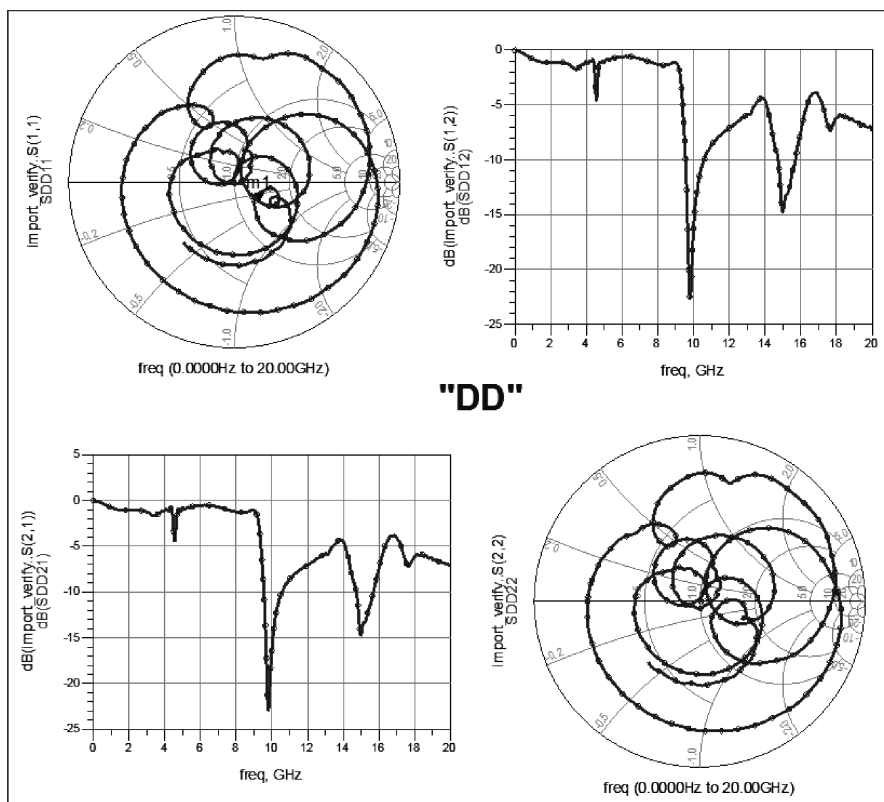


图 2-29 结果对比

# 实例 3 传输线及三维连接器 TDR 仿真

### 【目的】

在设计高速电路版图时，只要信号遇到阻抗突变，就会发生发射。阻抗突变的地方可能在线末端或者是互连线拓扑结构发生改变的任何地方，如拐角、过孔、接插件、封装等地方。理解信号反射的源头和使用仿真、测量工具来预测反射大小，能够有助有信号完整性工程师进行版图阻抗控制。

针对无源物理互联结构的阻抗不连续性，可以使用时域反射计（TDR）进行测量。时域反射计由一部极高速示波器和一个阶跃信号发生器构成。图 3-1 所示的是其示意图，其原理是由阶跃信号发生器产生一个阶跃信号  $E_i$  到被测件，然后测量反射信号的大小  $E_r$ ，并记录延迟时间，反射系数  $\rho = E_r/E_i$ ，利用反射系数即已知的仪器特性阻抗  $Z_0$ ，即可推算被测件的特性阻抗  $Z = Z_0(1 + \rho)/(1 - \rho)$ 、杂散电容及杂散电感等值。一般主要的应用为测量电缆线断点的距离、印制电路板布线的特征阻抗、杂散电容及杂散电感值的估算及 IC 封装电气参数的抽取等。

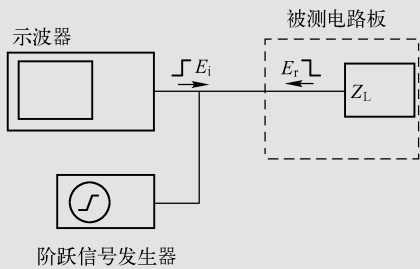


图 3-1 时域反射计原理图

使用仿真软件，能够方便快捷地对 PCB 走线以及 PCB 及高速连接器的配合情况进行 TDR 仿真，帮助工程师进行故障分析及定位。

在后续实验中，通过传输线及传输线和高速连接器的 TDR 仿真来说明其仿真流程。



## 3.1 PCB TDR 仿真实验

### 1. TDR 原理图仿真

(1) 在 ADS 中新建工程文件，命名为 TDR\_demo\_wrk。在本工程中，仅需要用到模拟/RF 库。注意，选择 Layout 的单位为 mil。

(2) 在主窗口中，新建一个原理图，并命名为 TDR\_demo。在原理图中插入双面板层叠

模型, 如图 3-2 所示。

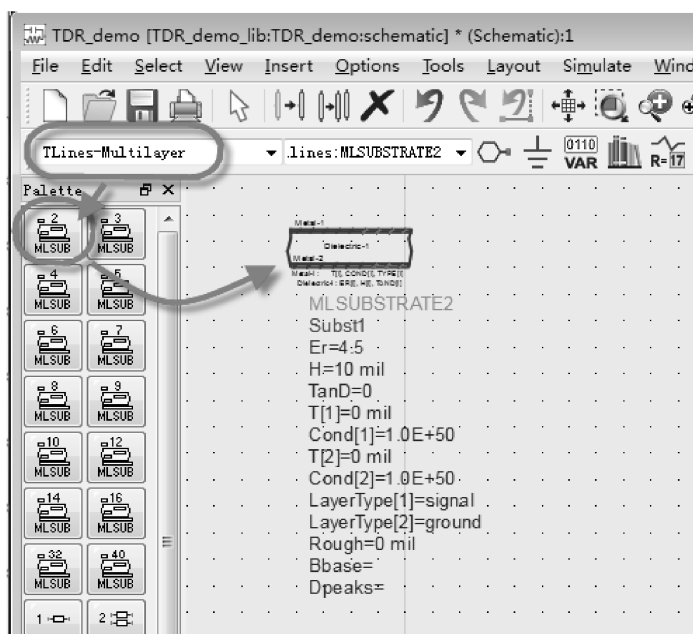


图 3-2 插入双面板层叠模型

修改其介电常数 ( $Er = 4.2$ )、介质厚度 ( $H = 28.8\text{mil}$ )、损耗角正切 ( $TanD = 0.02$ )、金属厚度 ( $T = 1.6\text{mil}$ ) 及电导率 ( $Cond = 5.8e7$ ), 如图 3-3 所示。

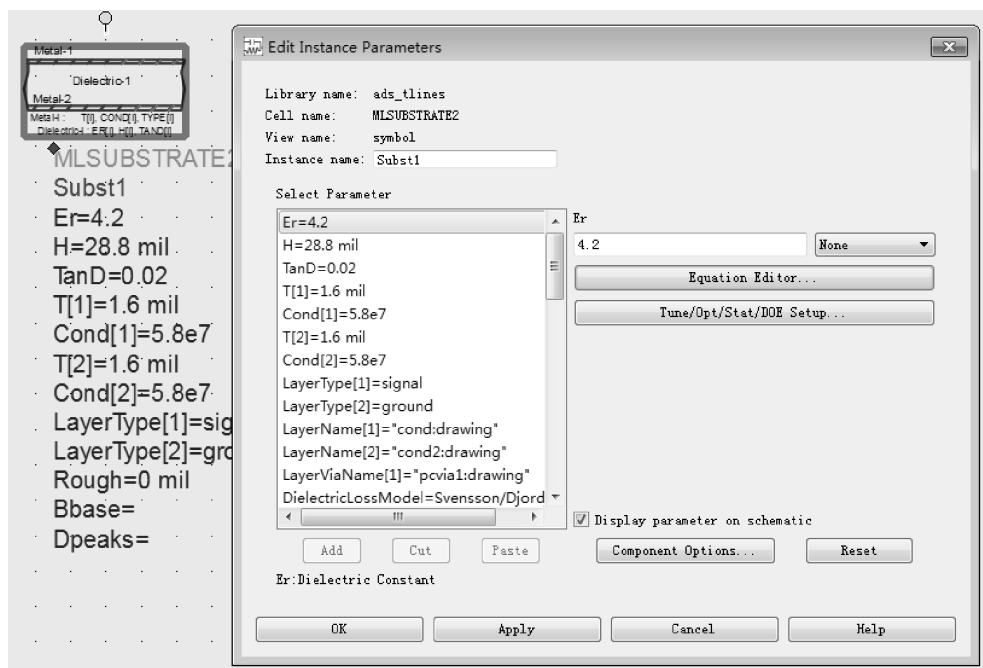


图 3-3 设置面板层叠模型参数

(3) 从多层传输线模型库 (TLines - Multilayer) 插入微带线模型并修改其长度及宽度, 如图 3-4 所示。

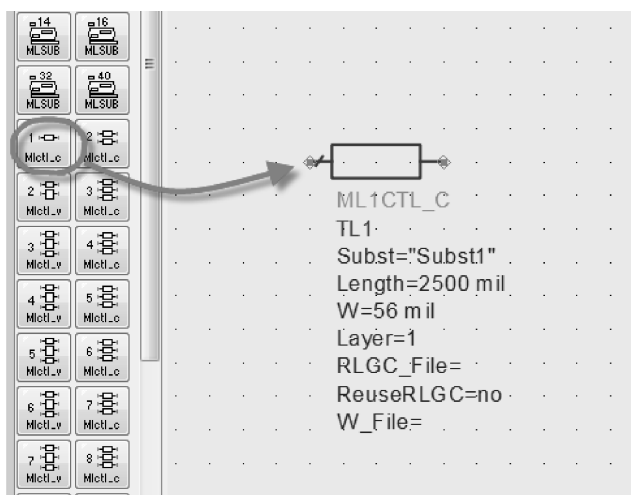


图 3-4 插入微带线模型

插入另外两段传输线模型, 参数设置如图 3-5 所示。

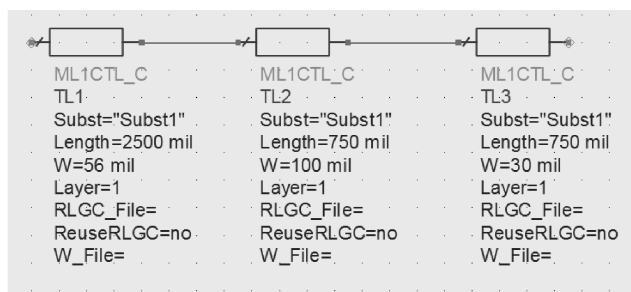


图 3-5 设置微带线模型

(4) 从时域源中插入脉冲源, 并设置参数, 如图 3-6 所示。

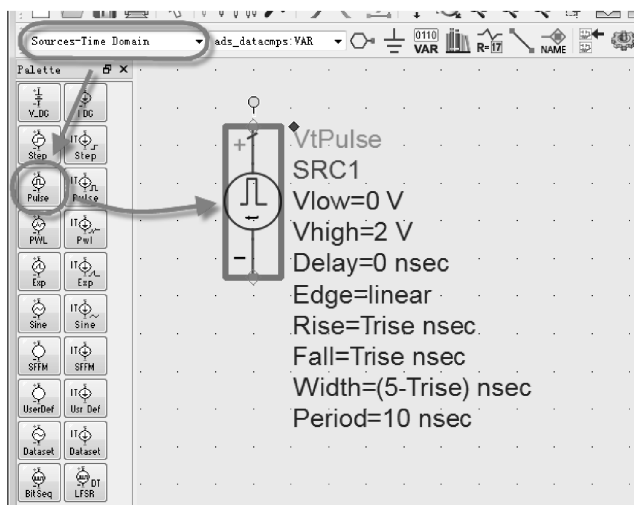


图 3-6 插入脉冲源模型



(5) 按图 3-7 所示将脉冲源、传输线进行连接，并在传输线前后串联 50Ω 电阻。

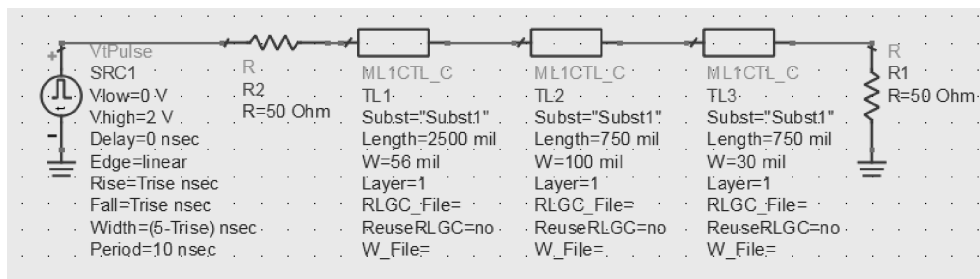


图 3-7 构建完整测试微带线模型

设置三个测试节点：脉冲源电压 Vscr、近端电压 V1 及远端电压 V2，如图 3-8 所示。

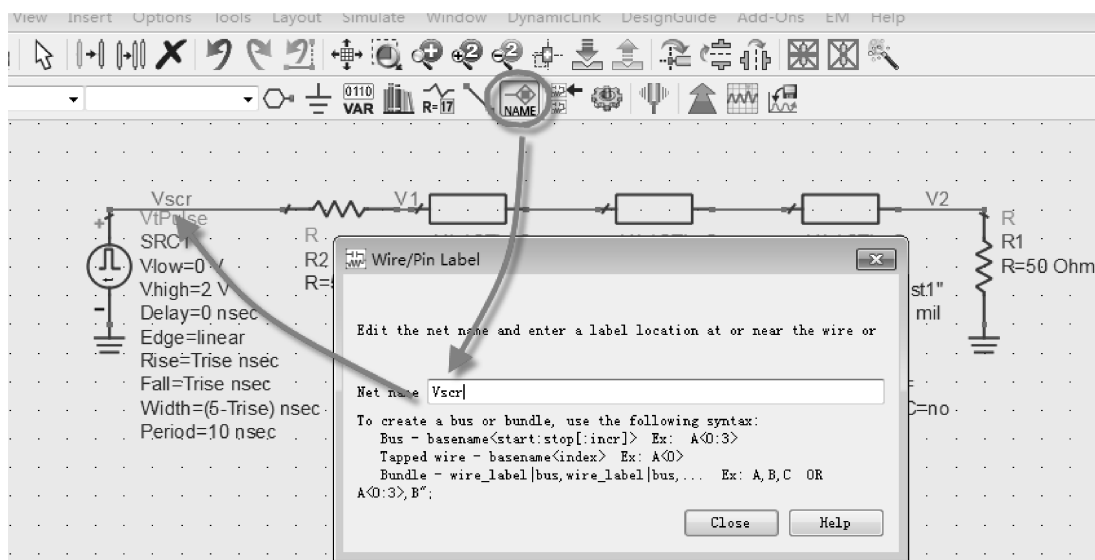


图 3-8 设置测试节点

(6) 如图 3-9 所示，加入瞬态仿真控制器，并设置最大采样时间为  $\text{Trise}/2$  nsec：

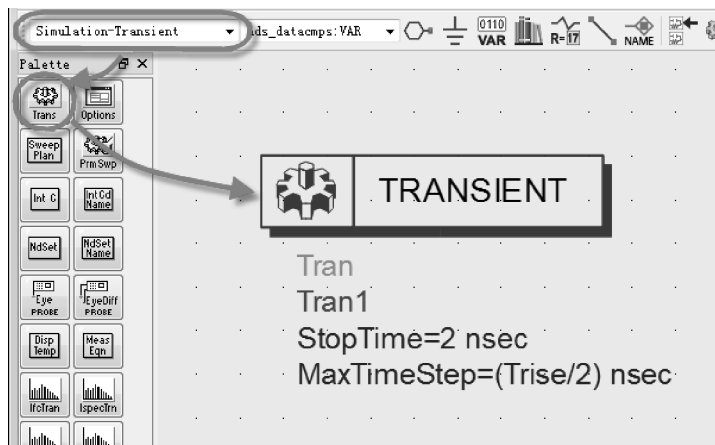


图 3-9 设置瞬态仿真控制器

如图 3-10 所示，插入变量 Trise，并定义其数值。

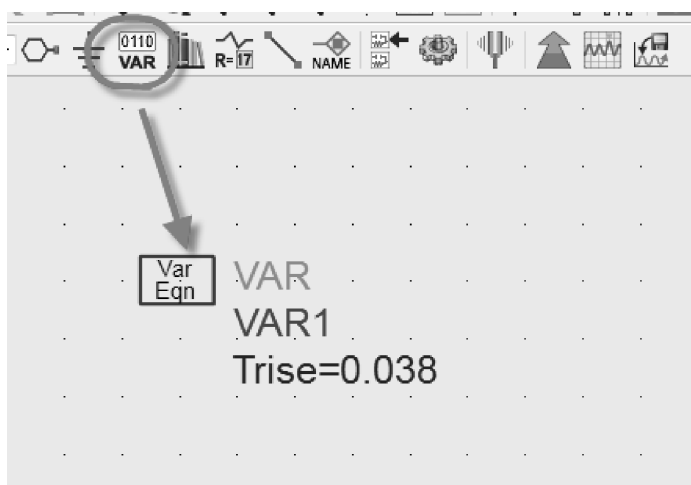


图 3-10 插入变量

完成后的原理图如图 3-11 所示。

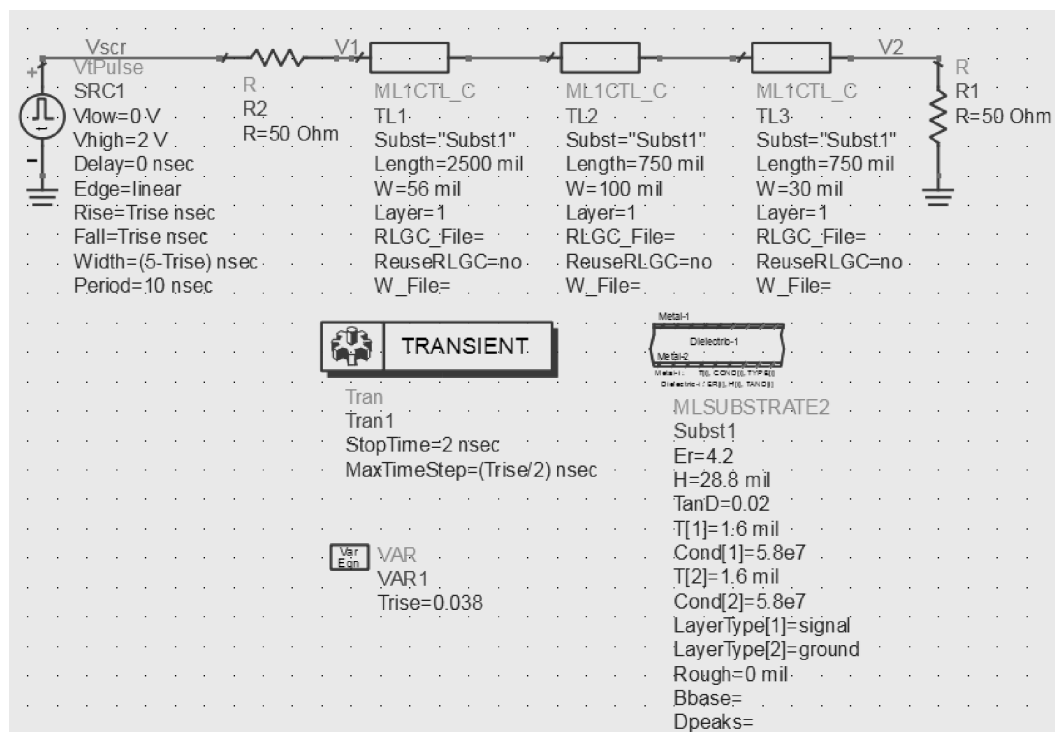


图 3-11 完整仿真原理图

(7) 运行仿真后，在结果显示中加入三个节点的电压波形进行观察，如图 3-12 所示。从反射信号 V1 可以看到信号链路中的容性及感性部分。

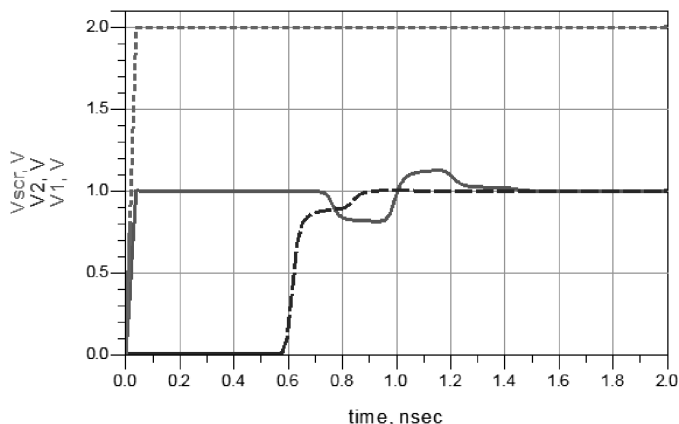


图 3-12 电压节点仿真结果

在原理图中能够使用传输线模型库中的基本元件构建信号路径，并使用瞬态仿真控制器来对信号走线的阻抗不连续性进行模拟。

## 2. 使用平面电磁场仿真器对传输线进行仿真

(1) 将原理图另存为 TDR\_demo\_EM，并删除掉仿真控制器等，仅保留传输线模型。在传输线模型前后加入两个端口（见图 3-13）。

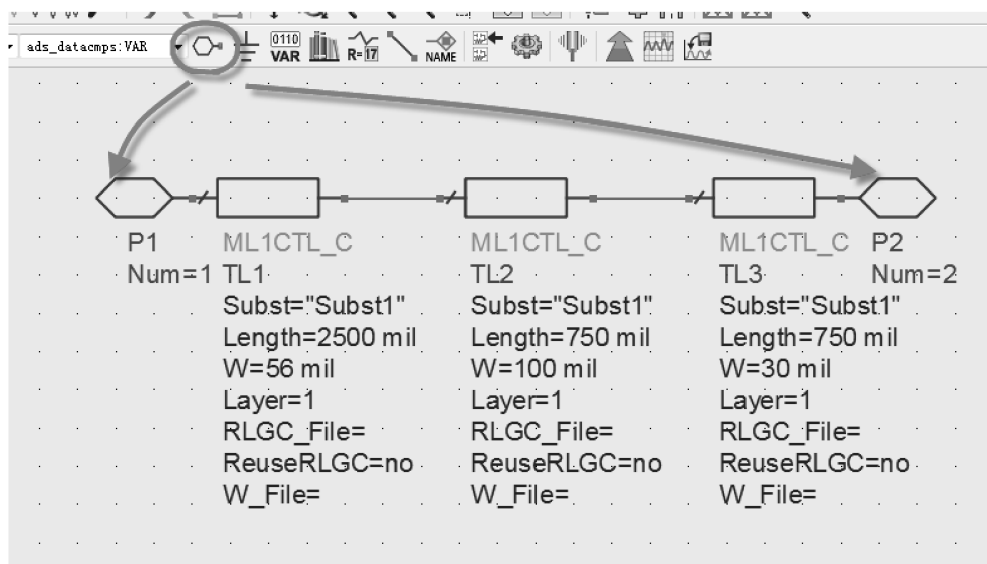


图 3-13 为传输线模型添加端口

在 Layout 菜单下选择 Generate/Update Layout，更新对应的版图（见图 3-14）。

在弹出的窗口中，直接单击 OK 按钮，ADS 软件会创建原理图对应的版图（见图 3-15）。

(2) 此时的版图，仅有传输线形状，并没有层叠文件与之对应。故需要添加层叠文件。单击层叠编辑器，系统会提示没有对应的层叠文件（见图 3-16）。

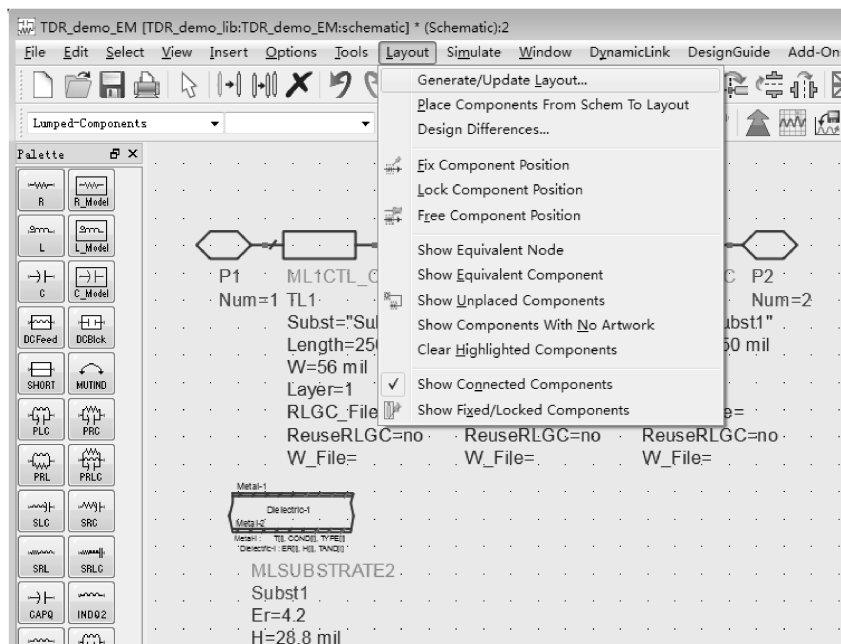


图 3-14 从原理图创建版图

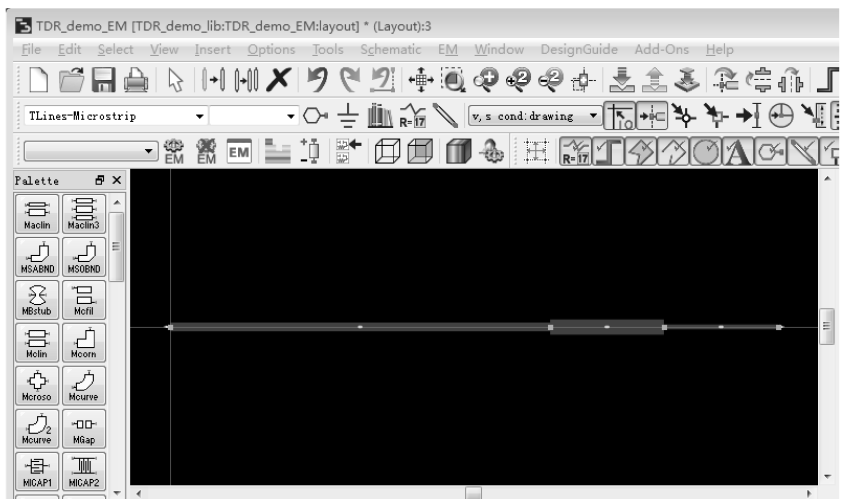


图 3-15 新生成的版图

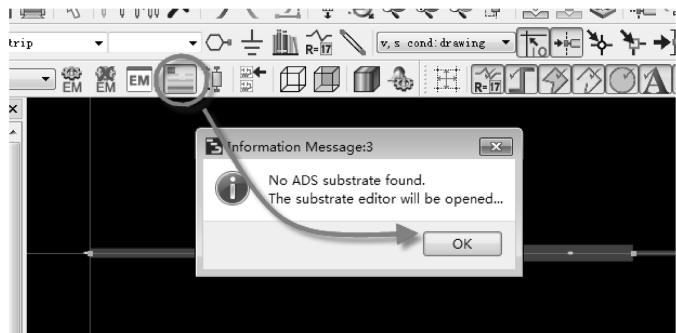


图 3-16 添加层叠文件

新添加一个层叠，命名为 substrate1。由于在原理图中已经定义了双层传输线层叠，故在层叠编辑器中只需要导入原理图中的层叠设置即可（见图 3-17）。

从 TDR\_demo\_EM 的原理图文件中导入层叠信息（见图 3-18）。

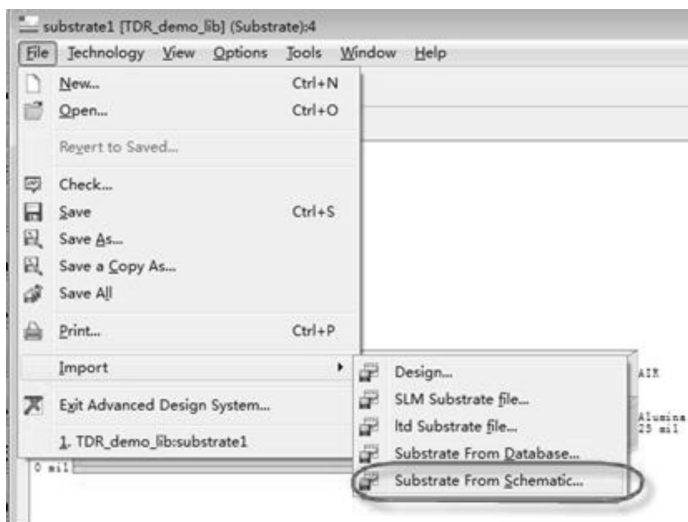


图 3-17 从原理图导入层叠文件

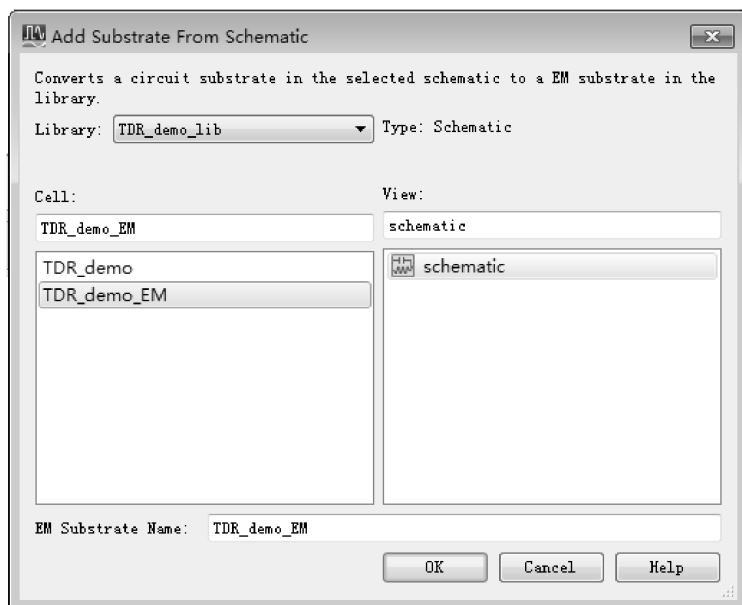


图 3-18 选择对应层叠文件

创建好的层叠如图 3-19 所示。将鼠标放到介质或金属层上可以观察设置的参数。

(3) 回到版图窗口，在菜单栏中选择电磁场仿真设置（见图 3-20）。

由于仿真对象的电尺寸较小，不考虑辐射情况，可以选择仿真器为 Momentum RF 以提高仿真速度。设置仿真频率从直流至 5GHz（见图 3-21）。

在模型选项卡中选择“仿真开始后创建对应 EM 模型”，保存 EM 仿真设置后，开始进

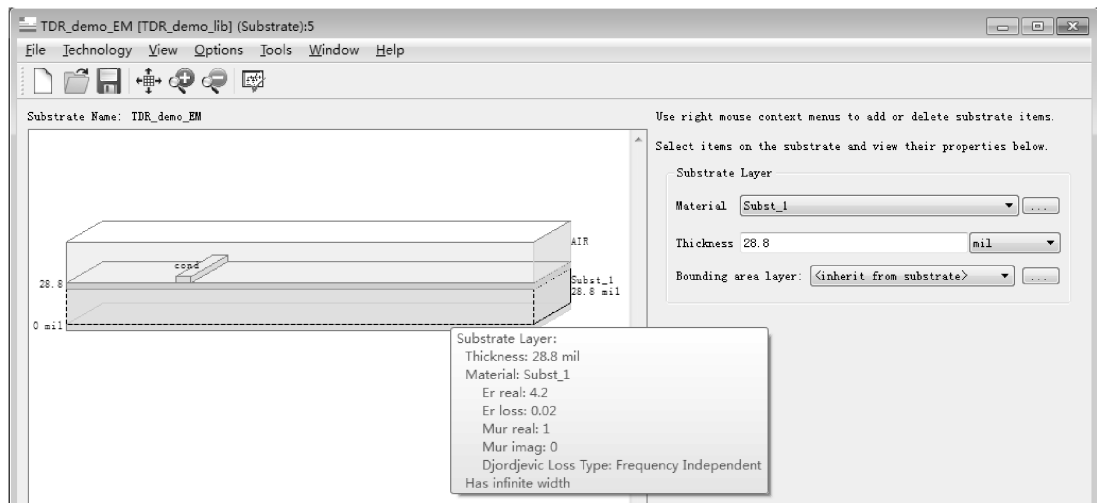


图 3-19 观察层叠文件



图 3-20 选择电磁场仿真设置

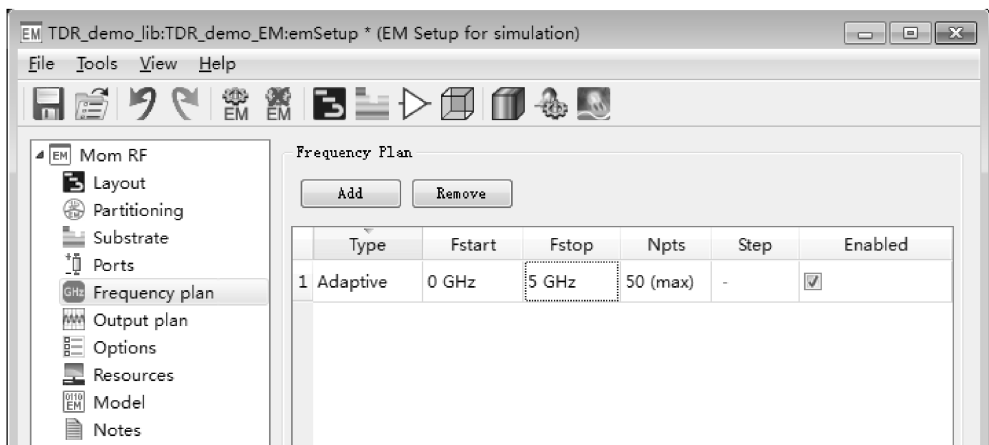


图 3-21 设置仿真频率

行电磁场仿真（见图 3-22）。

(4) 新建一原理图，命名为 TDR\_EM\_cosim，如图 3-23 所示，在 ADS 主窗口中使用鼠标将选中 TDR\_demo\_EM 单元，拖入到新建的原理图中。此时系统会提示没有对应的符号，是否需要创建一个对应符号，选择 Yes。

在创建符号窗口中，Source View 选项选择 layout，Symbol Type 选项选择 Look - akike，适当缩放其大小以和原理图中其他元件匹配，单击 OK 按钮就可以创建基于版图外形的符号（见图 3-24）。

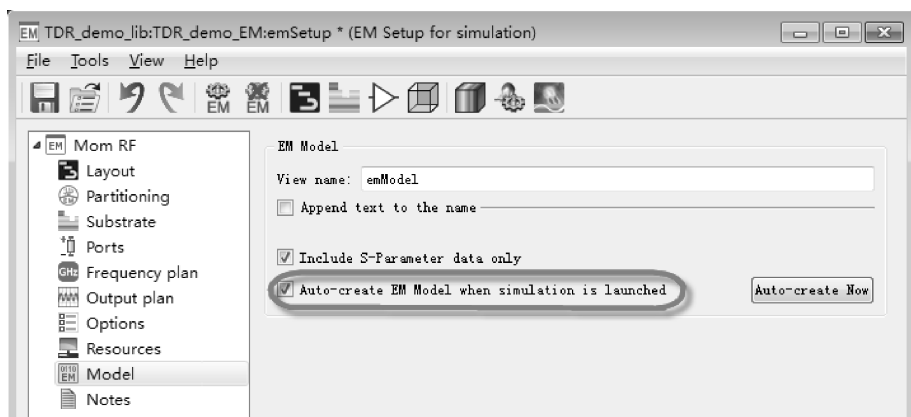


图 3-22 设置电磁场模型

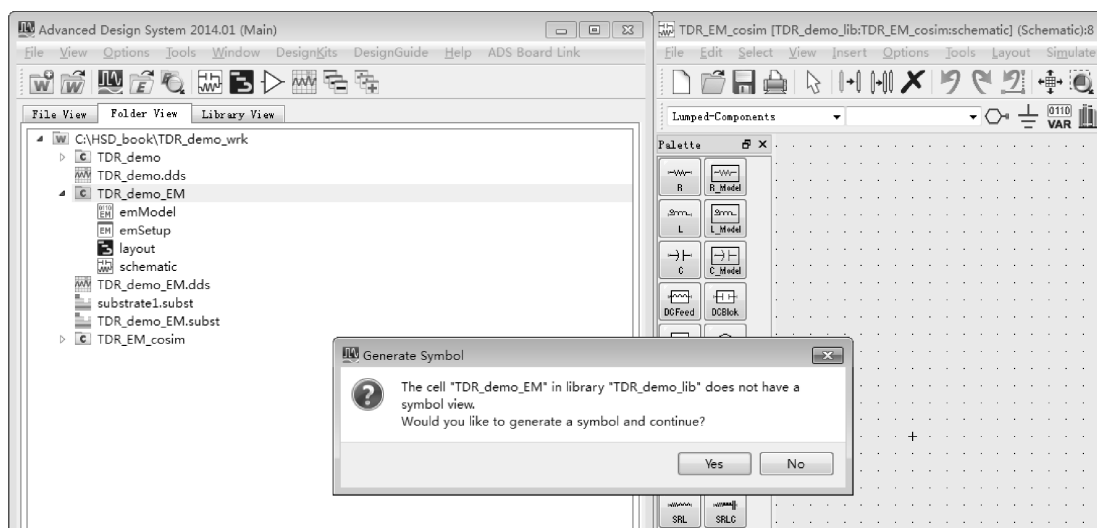


图 3-23 添加版图仿真单元至新建原理图

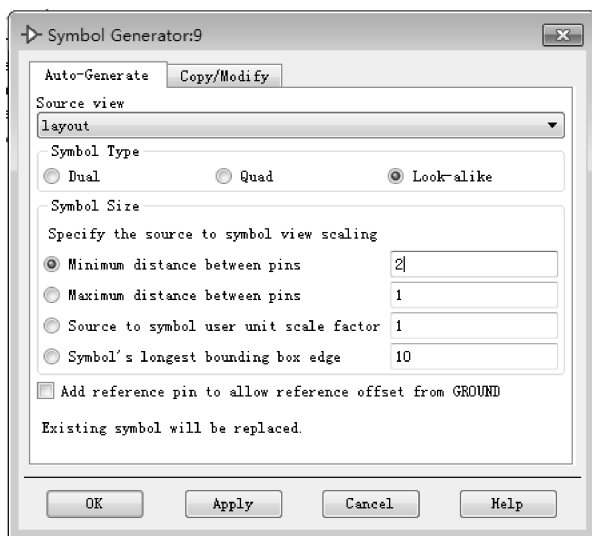


图 3-24 设置版图仿真单元符号

插入电磁场仿真元件后，再加入脉冲激励源、对应的源阻抗及负载阻抗，并加入观察节点（见图 3-25）。

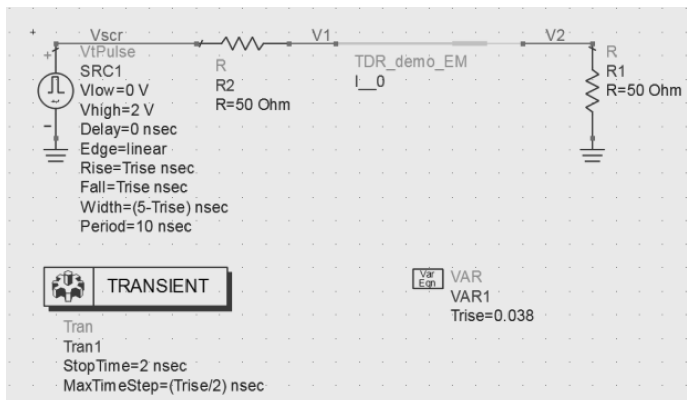



图 3-25 完整仿真原理图

(5) 单击原理图窗口中的  图标，选择电磁场仿真元件调用的视图为 emModel，即电磁场仿真结果（见图 3-26）。

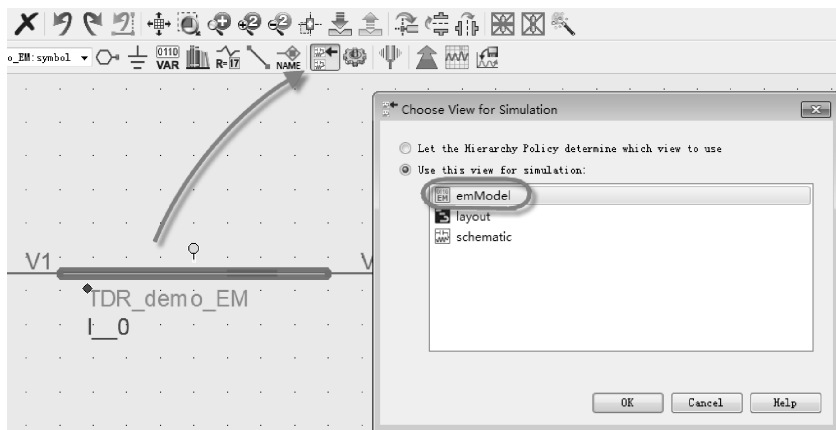


图 3-26 选择电磁场仿真模型

在原理图 Simulate/Simulation Settings 菜单中，对仿真的数据库进行设置，命名为 TDR\_EM\_5GHz\_cosim。单击 Simulate，运行仿真并观察仿真结果（见图 3-27）。

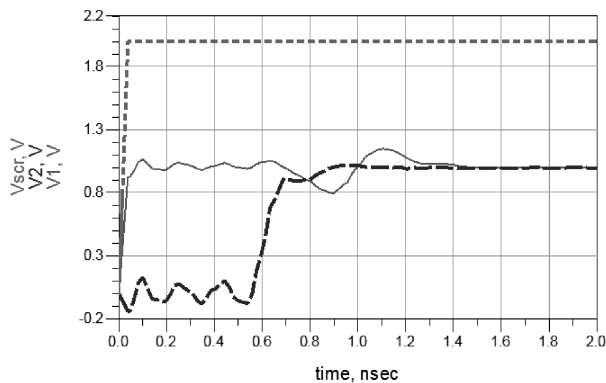


图 3-27 电压节点仿真结果



(6) 尝试将电磁场仿真元件的仿真频率改得高一些。在原理图中选中元件后单击 Push Into Hierarchy (见图 3-28)。

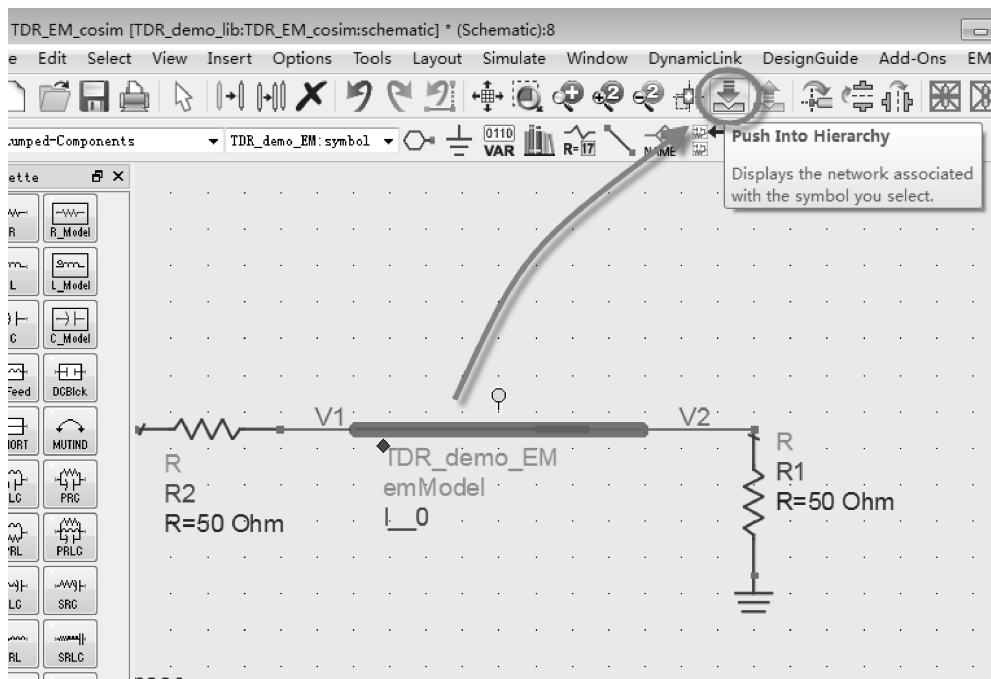


图 3-28 进入电磁场模型底层

对 emModel 设置进行编辑，如图 3-29 所示。

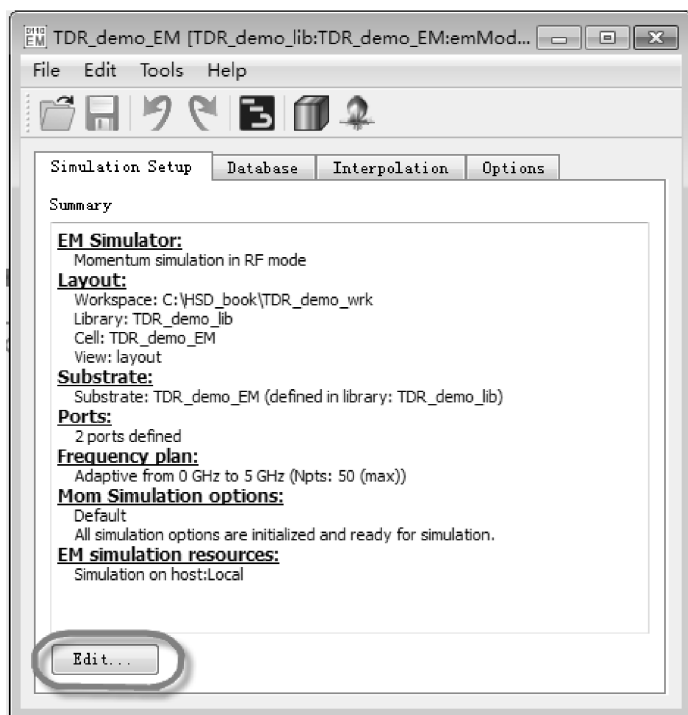


图 3-29 编辑 emModel 设置

将仿真频率上限改为 10GHz，如图 3-30 所示。

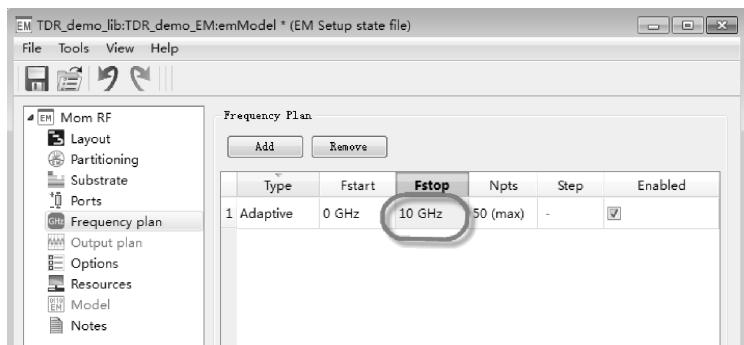


图 3-30 更改仿真频率

关闭设置窗口并回到原理图，在仿真设置中更改仿真数据库为：TDR\_EM\_10GHz\_cosim（见图 3-31）。



图 3-31 更改仿真数据库名字

仿真结果中纹波会比较小，如图 3-32 所示。在使用电磁场仿真器对传输线进行仿真时，需要确定仿真频率范围大于  $1/(0.35 \times \text{上升沿时间})$ 。对于 0.038 nsec 上升沿时钟的脉冲信号而言，对应的电磁场仿真最低频率需要满足大于  $0.35 * (1/0.038 * 10e-9)$  即 9.21GHz。

(7) 在结果显示窗口中编辑公式，显示仿真获得的传输线阻抗，如图 3-33 所示。Ref 是反射系数，为反射波形比上入射波形。

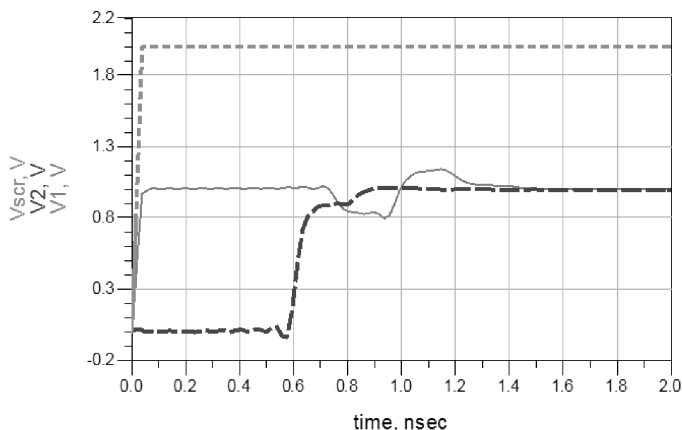


图 3-32 电压节点仿真结果

在数据显示窗口中插入新编辑的公式，如图 3-34 所示。

可以获得更加直观的传输线阻抗，如图 3-35 所示。

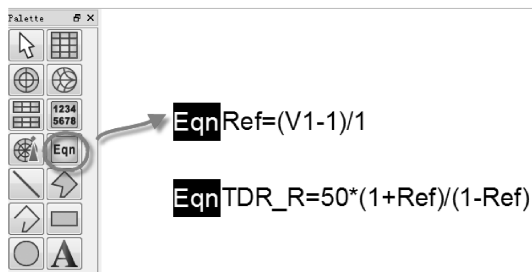


图 3-33 编辑 TDR 阻抗公式

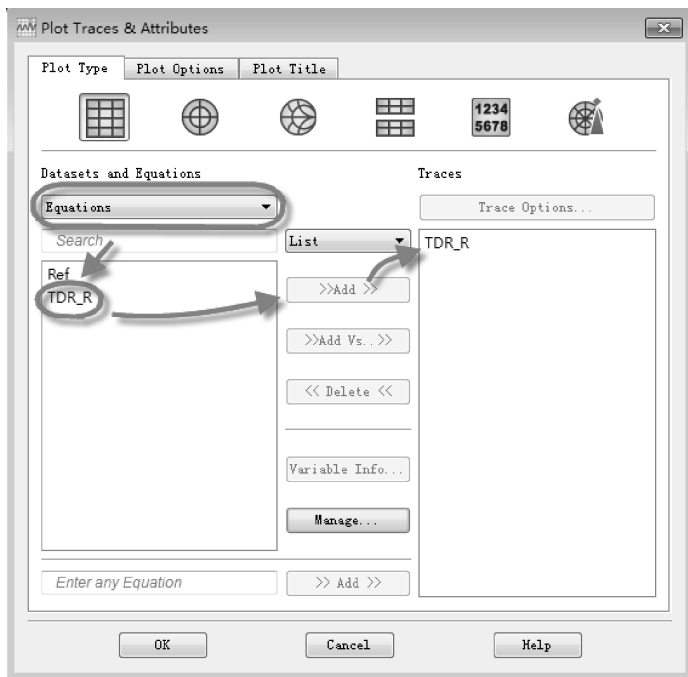


图 3-34 插入 TDR 阻抗公式

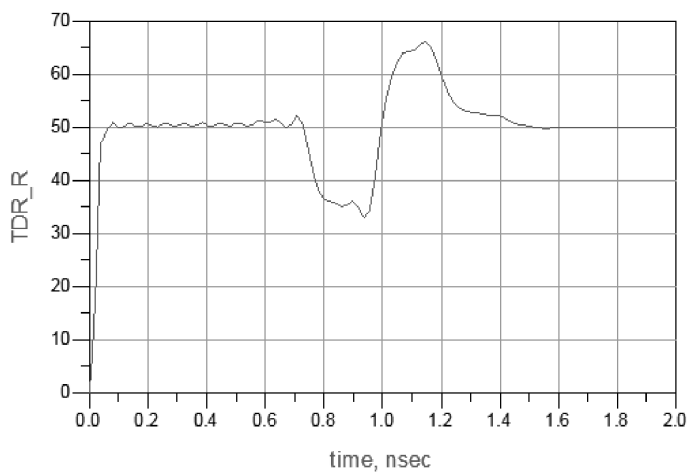


图 3-35 传输线阻抗仿真结果

在本小节中，使用 Momentum 对传输线版图进行电磁场仿真，并加入阻抗计算公式，获得传输线的 TDR 阻抗，对传输线的阻抗进行直观显示。



## 3.2 PCB 及连接器 TDR 仿真实验

在实际的信号通道中，信号不仅在 PCB 上进行传输，也常常通过接插件，在不同板间进行传输。在本例中，以使用 Samtec YFT/YFW 系列连接器为例，来说明如何进行整个传输通道的 TDR 仿真。图 3-36 所示为 YFT/YFW 系列连接器分别表贴在 PCB 上及配合后的照片。

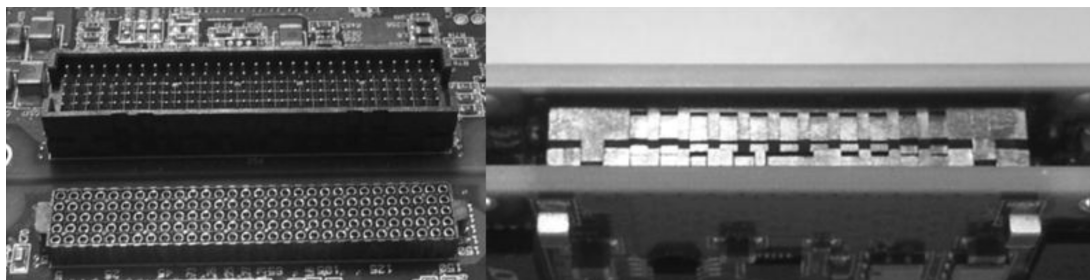


图 3-36 Samtec 连接器

(1) 可以从 Samtec 官网下载 YFT/YFW 系列连接器的三维实体模型，将其导入 EMPro。附件中对应的模型为 YFS-20-03-X-05-SB.step 及 YFW-20-07-X-05-SB.step。图 3-37 为 YFS（母头）系列连接器的正反面三维视图。可以看出，其使用焊球和 PCB 上的焊盘进行电气连接。

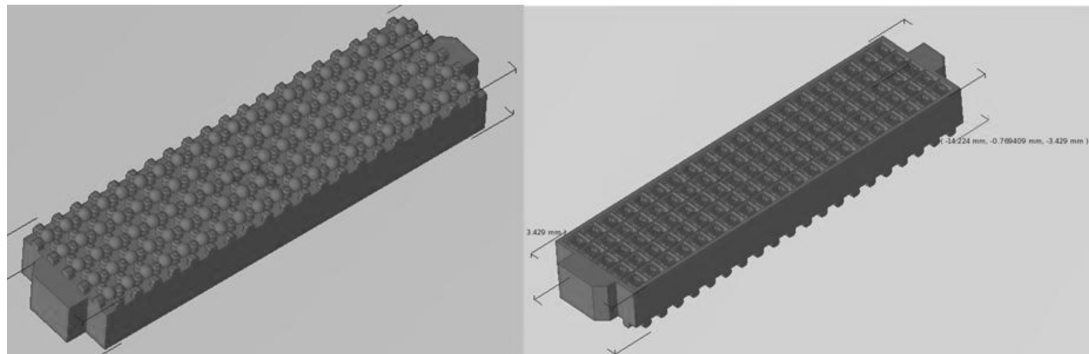


图 3-37 Samtec YFS 系列连接器三维视图

图 3-38 所示为 YFW（公头）系列连接器的三维视图。模型精细描述了连接器的机械结构。图 3-39 中为 YFS 系列连接器中的插针套筒以及与其配合的焊球。

(2) Samtec 公司在其官网给出了 YFT/YFW 系列连接器的推荐 Layout 焊盘图，其定义了焊盘阵列尺寸及过孔位置等，如图 3-40 所示。

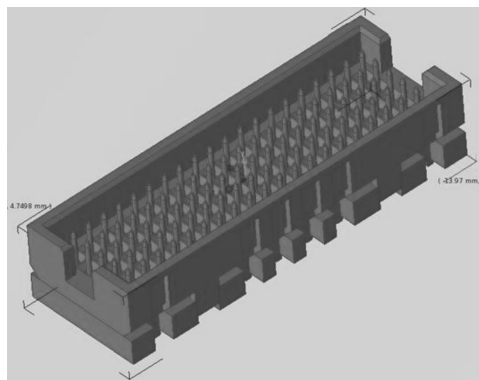


图 3-38 Samtec YFW 系列连接器三维视图

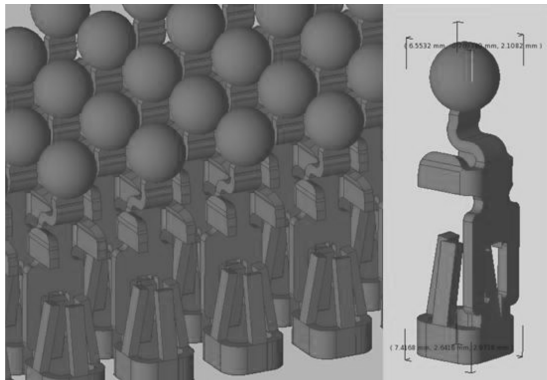


图 3-39 焊球模型三维视图

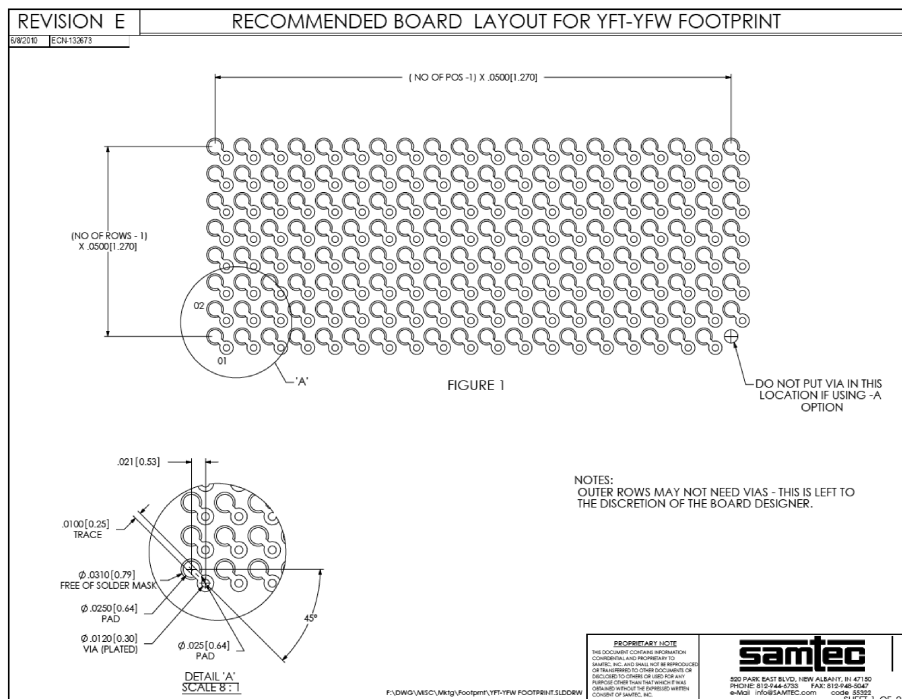


图 3-40 推荐 Layout 图

在 ADS 中依照参考设计进行焊盘及过孔阵列设计（见图 3-41）。附件 Samtec\_footprint\_wrk 工作区中的设计 YFT\_FYW\_footprint\_full 已经完成了相应的参考设计。

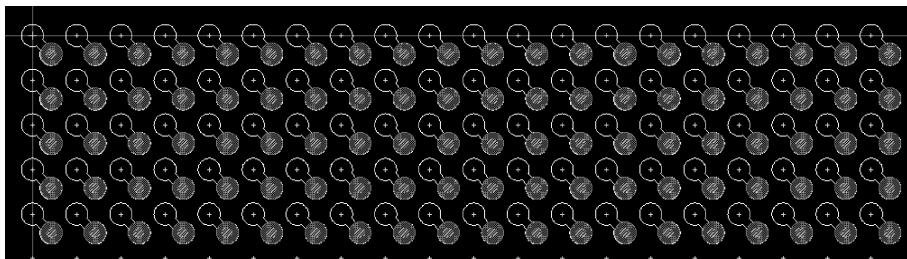


图 3-41 在 ADS 中实现焊盘及过孔 Layout

在 ADS 的层叠编辑器中，按照真实的层叠情况进行设置（见图 3-42）。

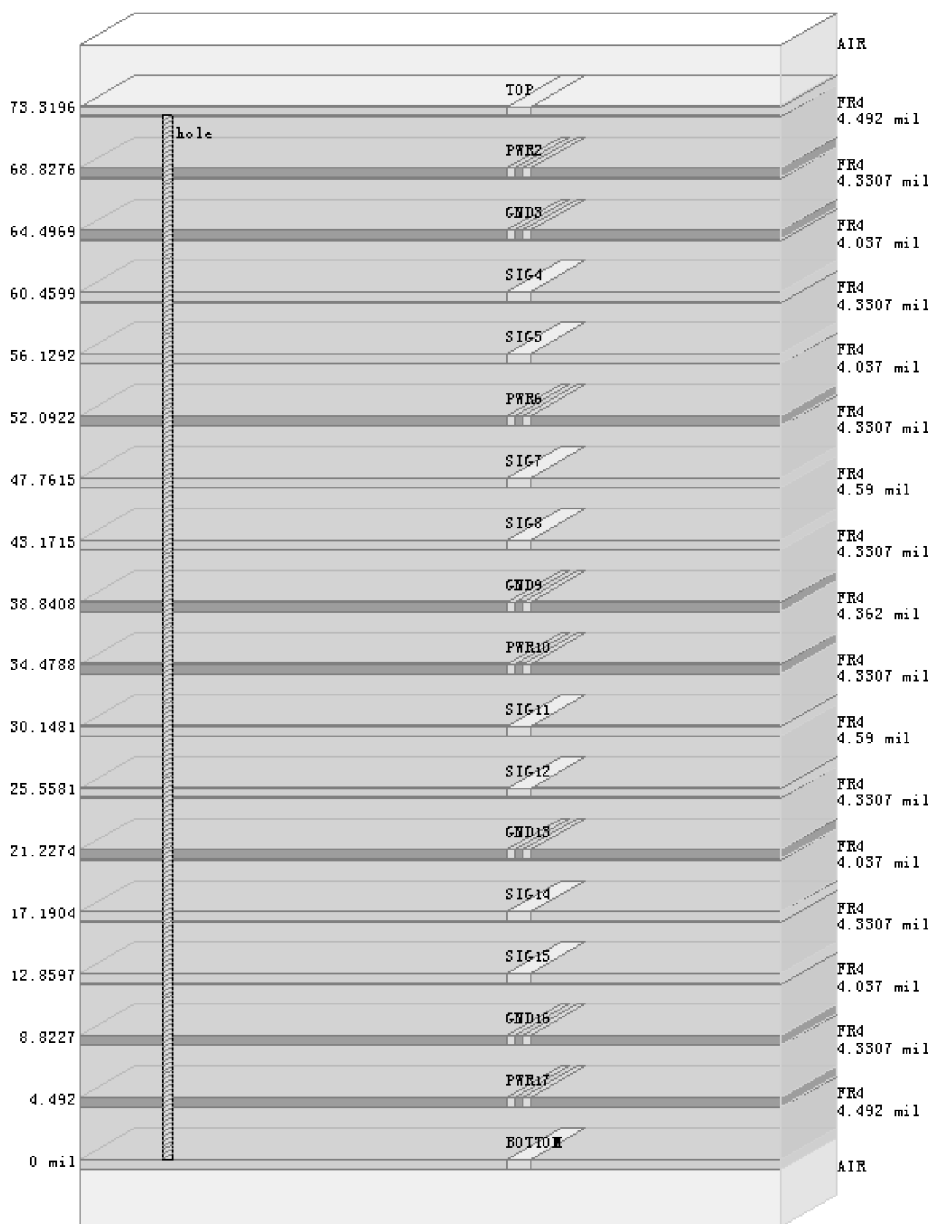


图 3-42 Layout 层叠设置

(3) 连接器通过焊球焊接到 PCB 上时，焊球会有一定的形变。在中 ADS 中可以使用形变的焊球模型准确表征实际焊球。

在 ADS 中可以直接调用 EMPro 生成的三维焊球模型。在 ADS 主菜单中，选择 Design-Kits/Manage Libraries，添加 EMPro 安装目录下的 Example/BasicComponentsForADS 文件夹做为 ADS 的库（见图 3-43）。

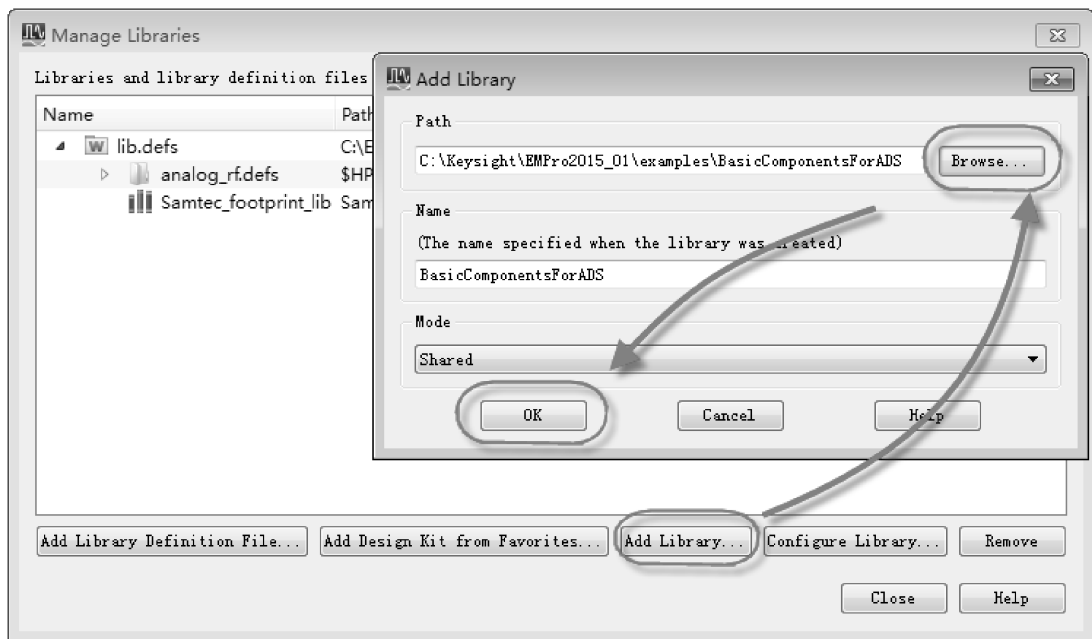


图 3-43 添加 EMPro 基本元件库

将 Solder Ball Array 拖入截取的部分版图图中（见图 3-44）。

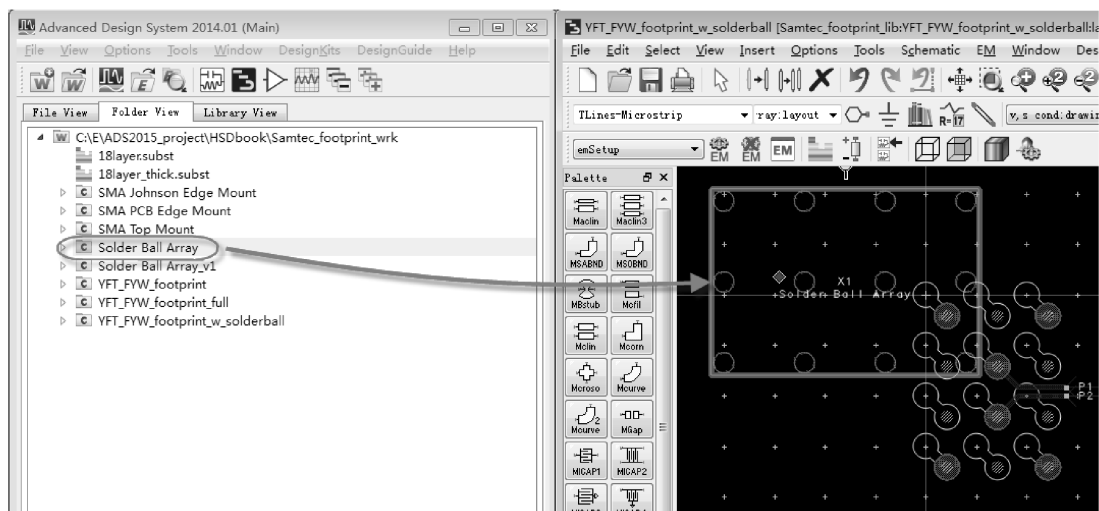


图 3-44 加入过孔阵列至 PCB Layout

如图 3-45 所示设置焊球参数。将焊球和 Layout 进行配合。使用焊球阵列模型，可以描绘受到应力变形后的焊球形状，如图 3-46 所示。

在 3D 预览器中确定焊球和焊盘的配合后，在 EM 设置中可以直接将焊盘和电路板的实体模型及材料、端口、边界调节等导出至 EMPro。在导出过程中，需要设定 EMPro 的安装路径。导入 EMPro 的模型如图 3-47 所示。

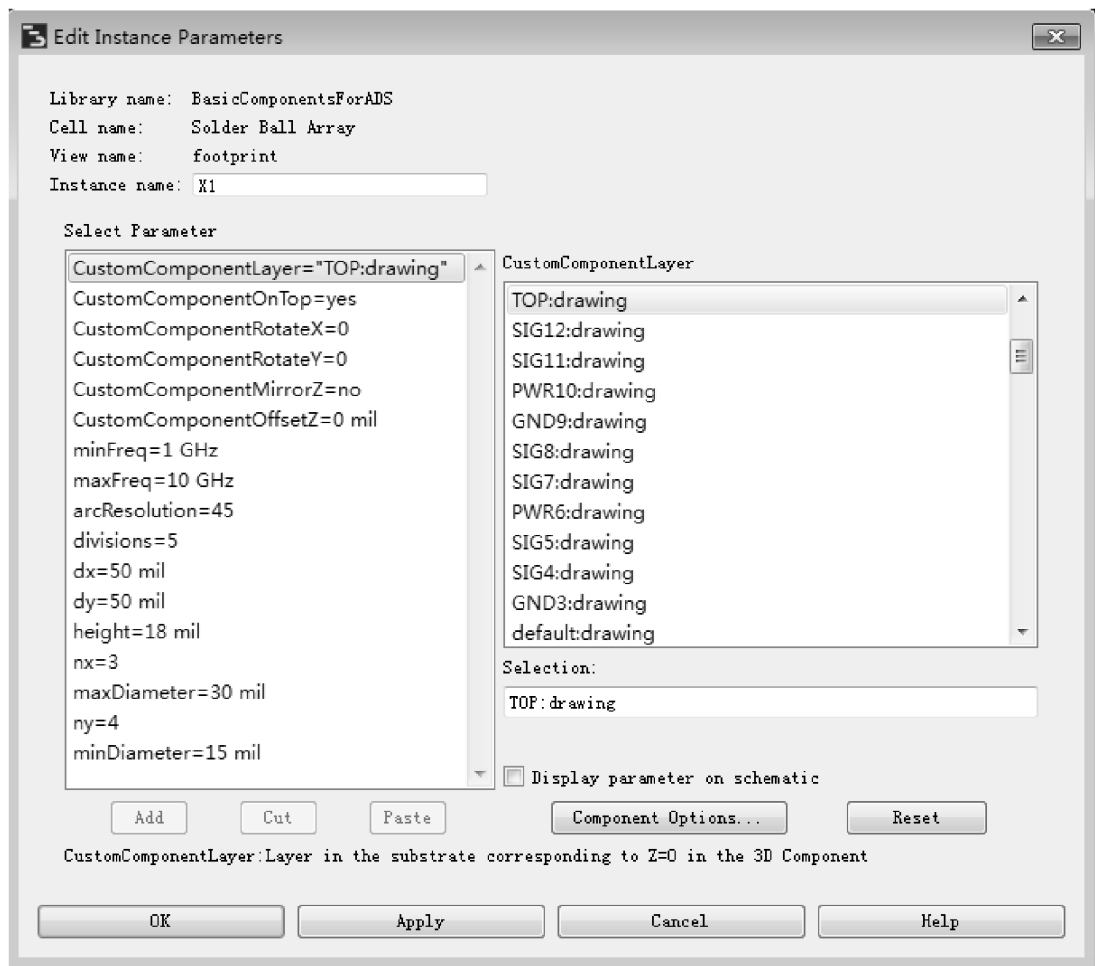


图 3-45 焊球阵列参数设置

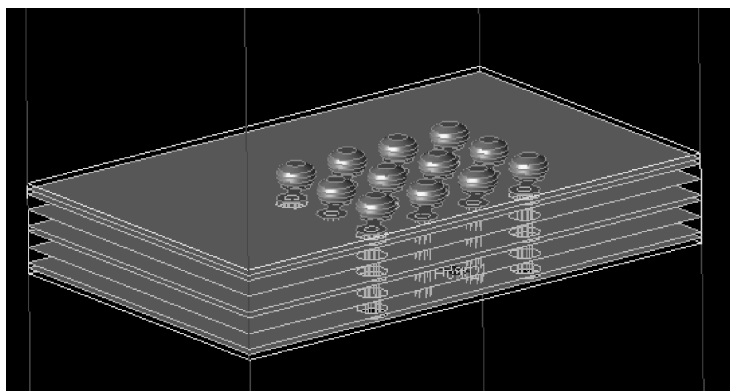


图 3-46 形变的焊球模型



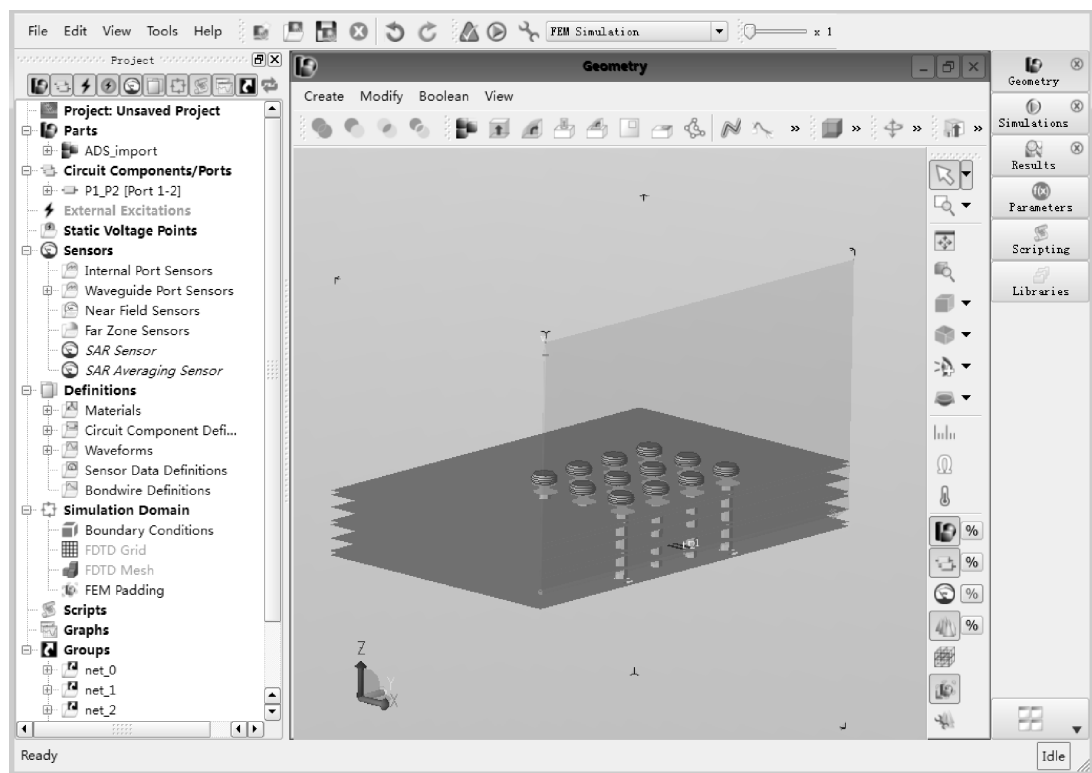


图 3-47 导入 EMPro 的 PCB 及焊球模型

(4) 在 EMPro 中, 将 YFT/YFW 系列接插件进行配合, 并进行对应材料设置, 如图 3-48 所示。接插件在使用时, 每一个信号旁边会有参考地。在进行 TDR 仿真时, 不需要对所有信号传输路径进行仿真, 选取其中的几对信号及参考地进行评估即可。截取模型的一部分作为仿真对象, 如图 3-49 所示。

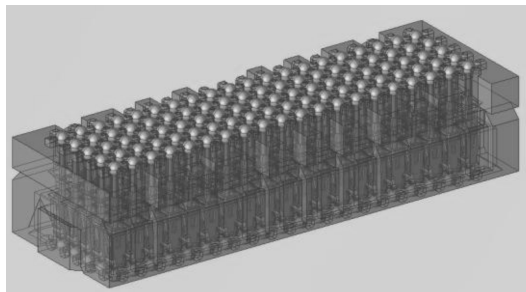


图 3-48 配合后的连接器

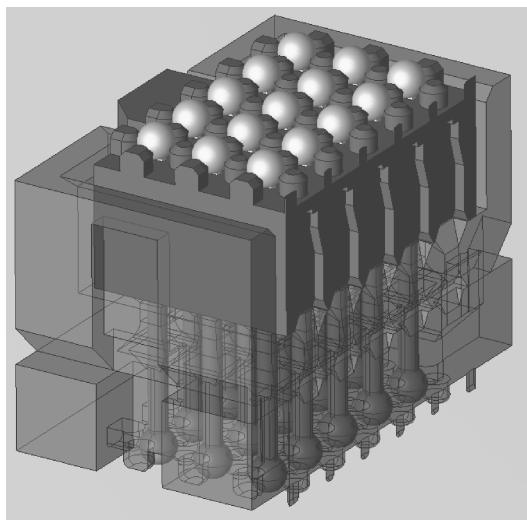


图 3-49 截取部分连接器模型

为实现连接器和 PCB 及焊球的精确配合，在 EMPro 中使用 locator 来进行精确定位，如图 3-50 所示。通过分别在 PCB 中焊球上及连接器上设置 locator，再使其重合，能够实现三维实体模型的准确配合。图 3-51 所示为发射及接收 PCB 以及中间的连接器的。

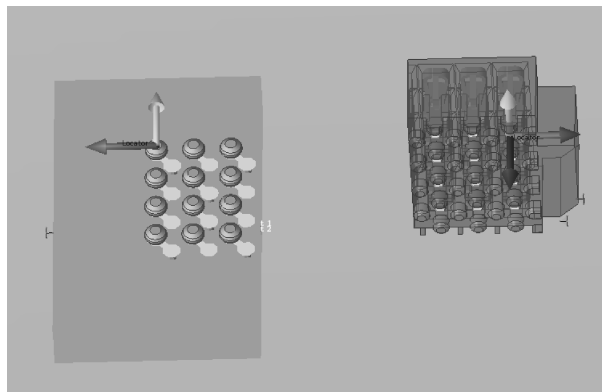


图 3-50 使用 locator 对连接器及 PCB 进行精确定位

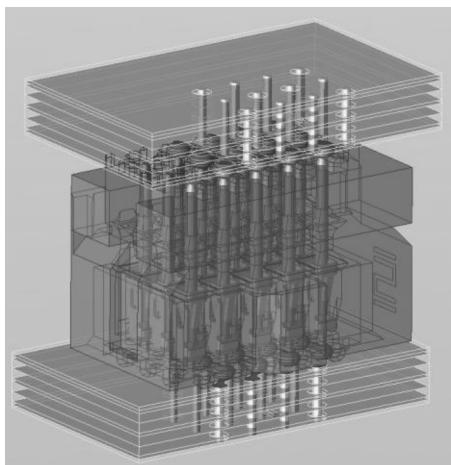


图 3-51 完整信号传输通道

(5) 工程文件 YFS\_YFW\_w\_PCB.ep 为设置好的示例文档。在 EMPro 中直接使用 TDR 阶跃信号进行激励，可以获得整个信号传输路径的阻抗。由于信号以差分形式传输，故特性阻抗应该在  $100\Omega$  左右。但是从 TDR 仿真结果可以看到，连接器引入了较强的感性及容性，如图 3-52 所示。

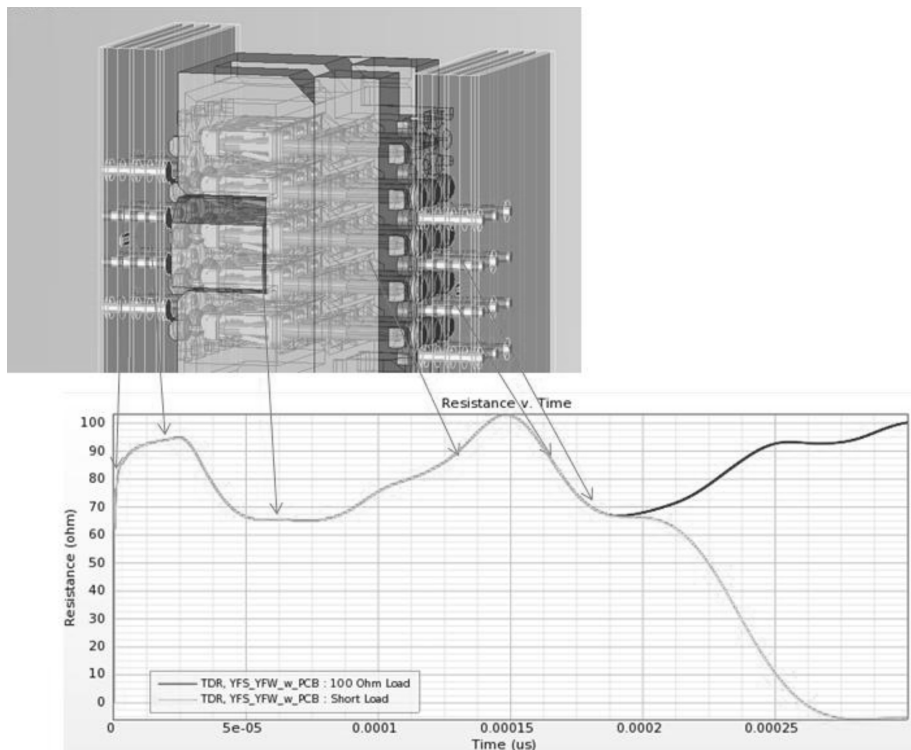


图 3-52 连接器 TDR 仿真结果

仿真时，分别设置 PCB 上的接收端口为  $100\Omega$  负载及短路，能够非常明确地对 TDR 仿真的终点进行定位。

#### 【总结】

在本实例中，通过传输线及传输线和高速连接器的 TDR 仿真来说明 TDR 仿真的基本原理及仿真流程。设计师能够使用 TDR 仿真器对信号传输通道的阻抗匹配情况进行直观分析和故障定位。

## 实例 4 基于测量的通道建模

### 【目的】

使用测试仪器对已制作完成的物理通道进行测试，再使用 ADS 软件建立通道的仿真模型，并使用 ADS 软件强大的优化功能，调整通道模型参数，使软件仿真结果逼近通道测试结果，完成通道模型的建立。通过这种方法，可以了解通道内各部分对整个通道的影响；同时，通过建立的完整通道参数化模型，可以方便地对通道各种物理参数进行调整，观察通道性能的变化，权衡各种物理设计参数以及板材特性参数指标，从而提高设计性能，降低成本，节省设计时间。

本实例中以图 4-1 所示的一款高速背板测试为例，在 ADS 软件中建立通道模型，并以测量得到的 S 参数为基准进行通道模型参数的优化，从而得到等效的通道仿真模型。

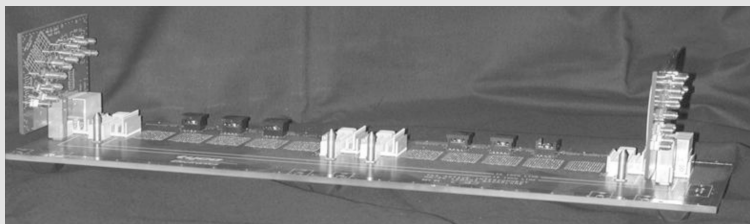


图 4-1 高速背板示例

本实例中使用 ADS2014.01 版本以及工程文件 ModelFromMeasurement.7zap。



### 4.1 通道测量结果显示

对物理通道进行测试一般使用网络分析仪或者 TDR 测试仪。网络分析仪具有宽带、高动态范围以及多端口同时测试的能力，本实例中使用网络分析仪测试得到的 S 参数（SnP 格式）。

启动 ADS 软件，解压 ModelFromMeasurement.7zap 工程。打开 S-parameter Simulation of Channel 目录下的 Channel\_Sparameter\_Simulation 原理图视图，如图 4-2 所示。

此原理图中，调用了 Channel\_16in.s4p 文件，是使用网络分析仪测试得到的 4 端口 S 参数文件。运行仿真，可以得到该通道 S 参数结果，如图 4-3 所示。

从 S 参数的仿真结果中，可以查看通道的频域响应，即通道的衰减、反射、串扰、耦合等结果。但是这样的结果对了解通道内部的特性是不够的。

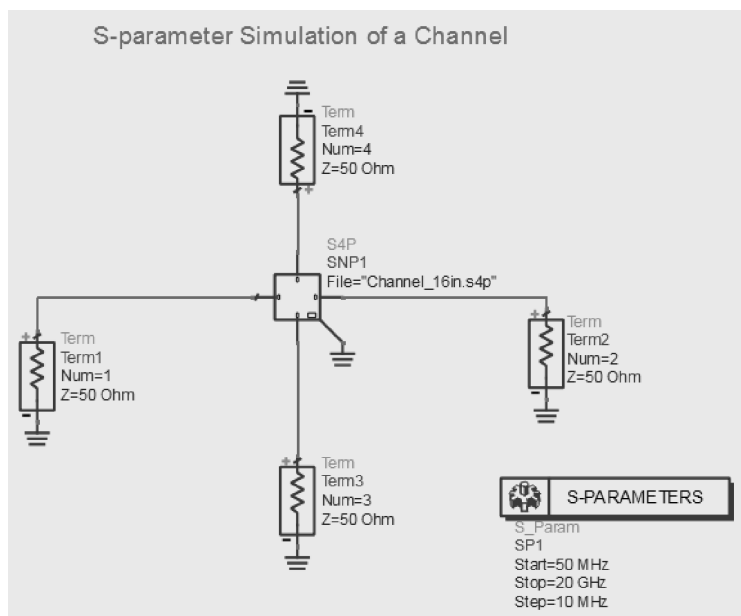


图 4-2 通道 S 参数仿真

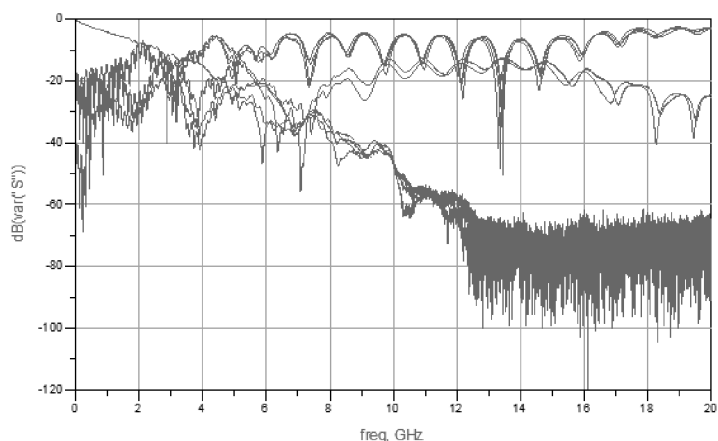


图 4-3 通道 S 参数仿真结果



## 4.2 通道时域特性仿真

打开 TDR Simulation of Channel 目录下的 TDR\_Simulation 原理图视图，如图 4-4 所示。

激励源使用的是脉冲源，在本实例中，主要关注上升沿以及随后一段时间内的反射信号，所以仿真时间只设定为 10nsec，运行仿真，显示仿真结果，如图 4-5 所示。

在数据显示窗口，使用函数 `TDR()`，可以将时域的电压响应转换为阻抗。在数据显示窗口 `TDR_ohms` 栏中显示阻抗曲线，如图 4-6 所示。

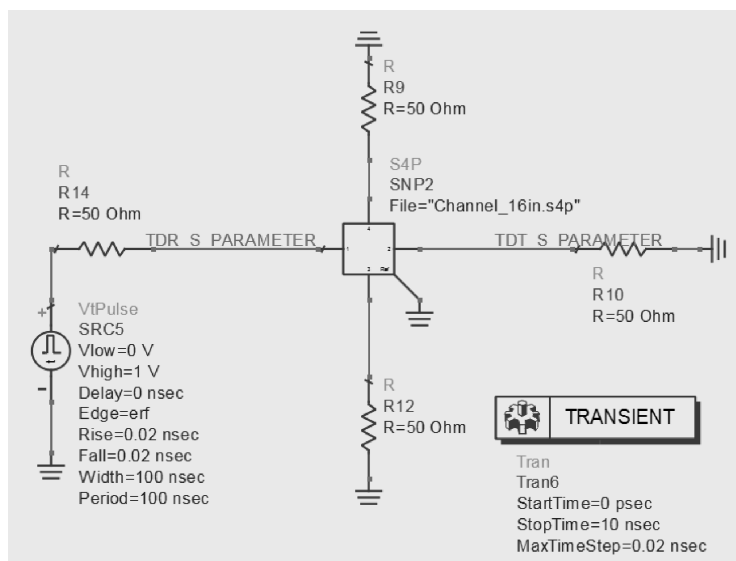


图 4-4 通道时域仿真模板

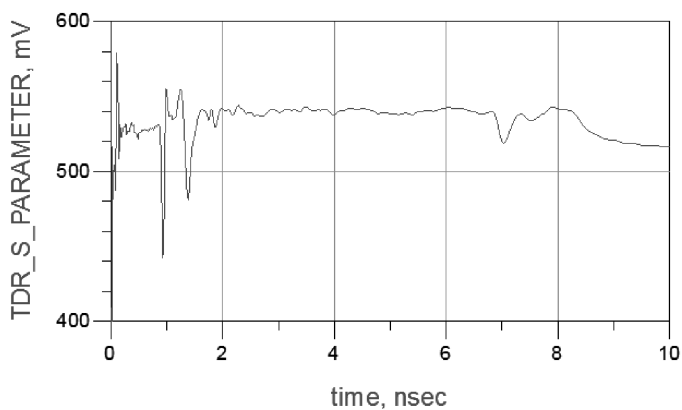
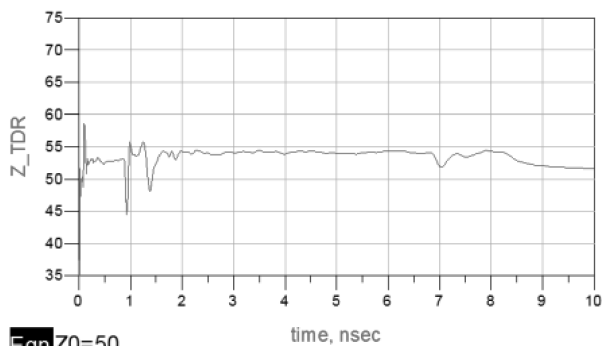


图 4-5 时域仿真结果



Eqn Z0=50

Eqn Z\_TDR=50\*((1+(TDR\_S\_PARAMETER-.5))/(1-(TDR\_S\_PARAMETER-.5)))

图 4-6 使用公式将时域反射电压转换为阻抗



### 4.3 建立通道模型

时域反射特性曲线对了解通道内特性非常有帮助。在时域反射曲线中，每一个阻抗的变化都是由于阻抗不连续造成的。通道中的接头、转接插件、过孔等都是造成阻抗不连续的原因。

根据时域响应并对比实物通道，可以为每一部分建立简单的仿真模型，如图4-7所示。

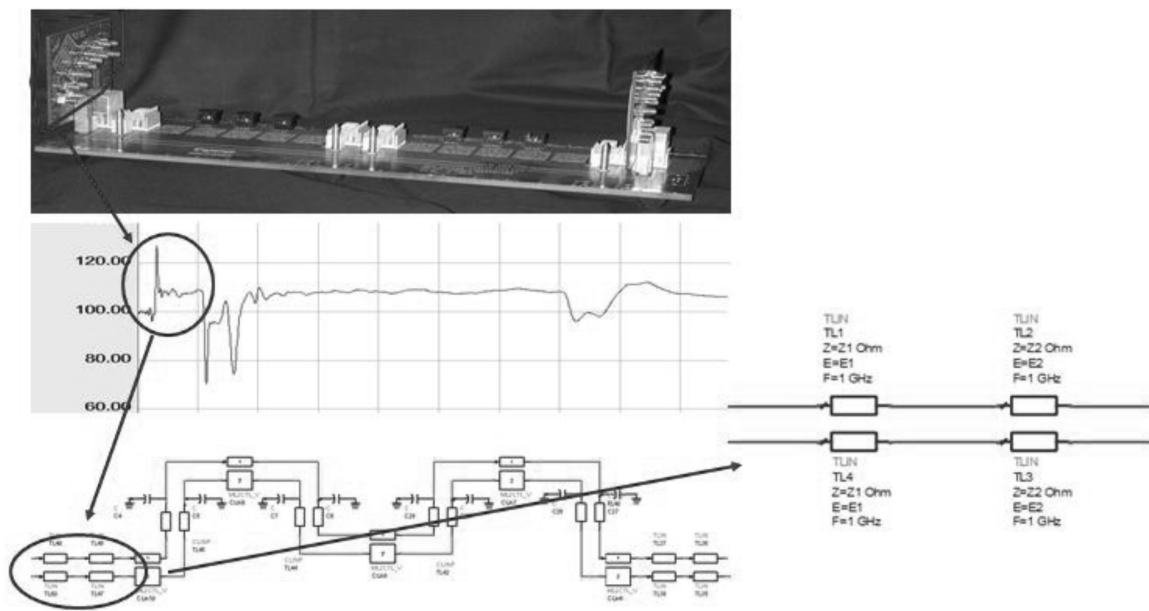


图4-7 高速背板子卡输入连接段的物理结构、时域反射响应及等效电路模型

子卡上的 SMA 接头及信号加载部分使用两段单端传输线构成。这两段传输线阻抗不同，从而在连接处造成阻抗不连续。原理图可见 Circuit Model Topology 目录下的 coax 原理图视图。子卡上的小段传输线可以使用 ADS 软件中的多层互连线模型库中的耦合传输线表征，Circuit Model Topology 目录下的 line\_card 原理图视图如图4-8所示。子卡上的过孔使用传输线加对地电容实现，详见 Circuit Model Topology 目录下的 via\_assembly 原理图视图，如图4-9所示。

子卡到背板的连接件使用传输线模型，详见 Circuit Model Topology 目录下的 connector 原理图视图，如图4-10所示。

背板上的过孔以及背板上的传输线均采用前述的拓扑结构，即使用耦合传输线加对地电容表征过孔；使用耦合传输线模型标准背板上的传输线。

本实例中的背板物理结构上是对称的，在模型上就可以根据这种对称性的特性，建立完全对称的通道模型结构。在 entire\_backplane\_ads\_design 原理图视图中，将已建立好的各单元模型级联起来，并将其模型参数设置为变量，为了方便比较，在原理图中还加入了通道测试数据，使用相同的激励源进行时域仿真，如图4-11、图4-12所示。

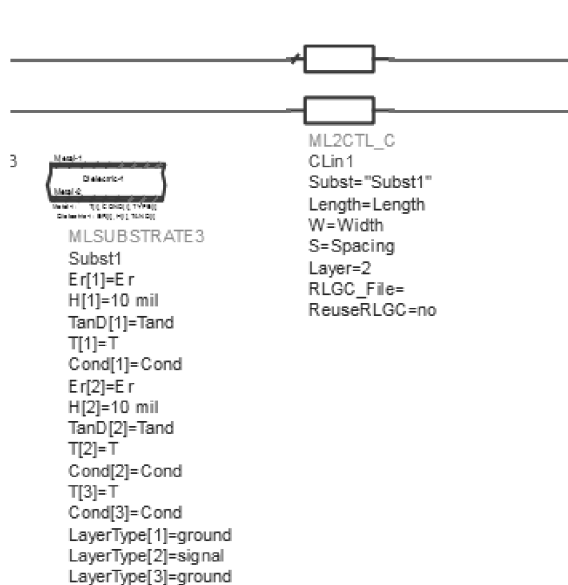


图 4-8 子卡上的传输线等效电路模型

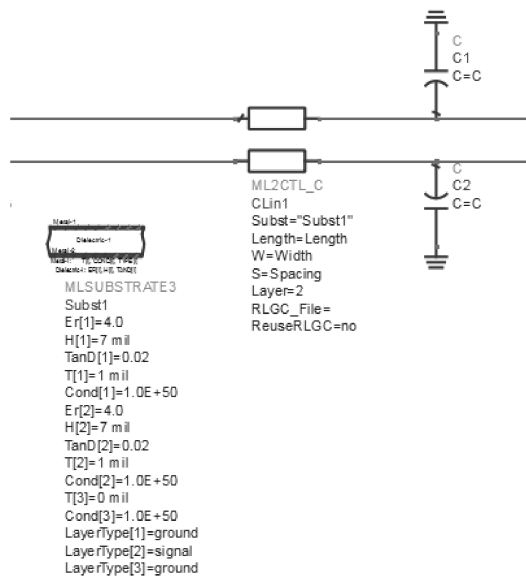


图 4-9 子卡上的过孔等效电路模型

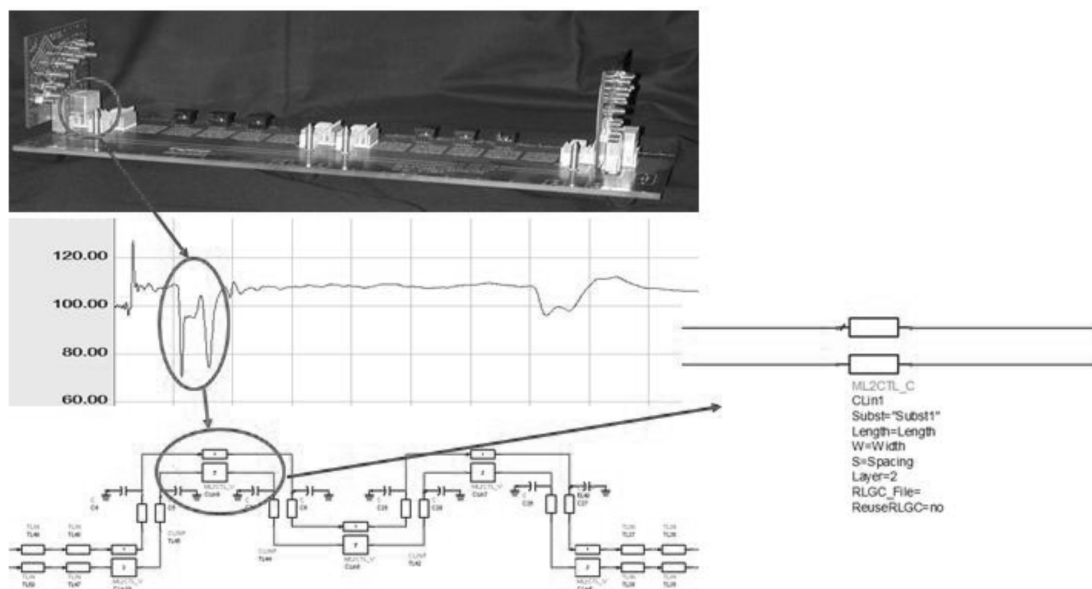


图 4-10 连接器物理结构、时域反射响应及等效电路模型

从仿真结果中可以看出，自建的通道模型和测试结果之间的差异是很大的。可以使用 ADS 软件中提供的调谐功能，调整各个模型的参数，使仿真结果逼近测试结果。图 4-13 所示显示调谐第一节 SMA 接头模型的 4 个参数，使得时域第一个反射点曲线基本吻合。



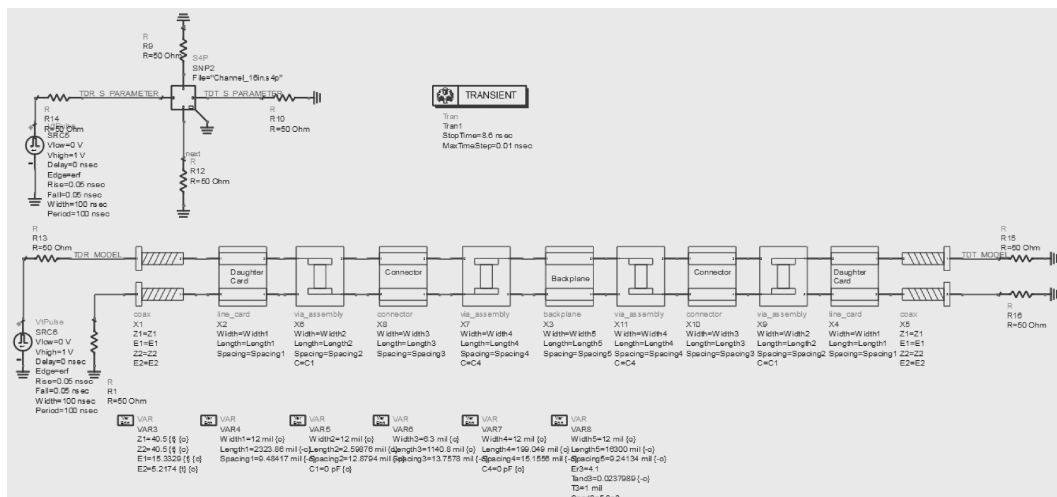


图 4-11 高速背板的等效电路模型

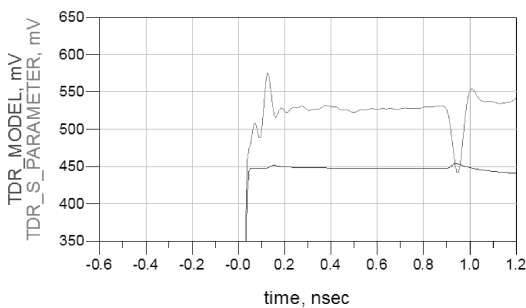


图 4-12 等效电路模型仿真结果（虚线）与实测结果（实线）对比

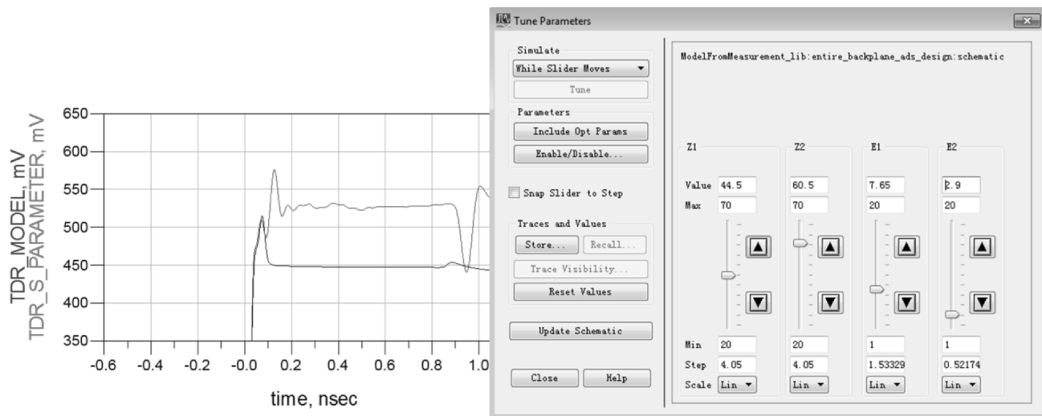


图 4-13 使用调谐功能调整输入端模型参数



## 4.4 通道模型的优化

使用调谐功能，非常直观。但是对于整个通道模型的几十个参数，调整起来效率较低。

这时候可以使用 ADS 软件提供的优化功能。

在使用优化功能时,需要设置优化控制器(设置优化算法、优化迭代次数、优化目标选择以及优化变量选择等)、优化目标(优化目标可以由表达式给出)、仿真控制器(通过什么样的仿真可以得到优化目标的值)以及设定优化变量(指定可以被优化的变量以及变量可优化的范围)。

在 Circuit Model Topology Optimization 目录下的 entire\_backplane\_ads\_design\_optimize 原理图视图中给出了使用优化功能进行通道模型参数拟合的过程,如图 4-14 所示。

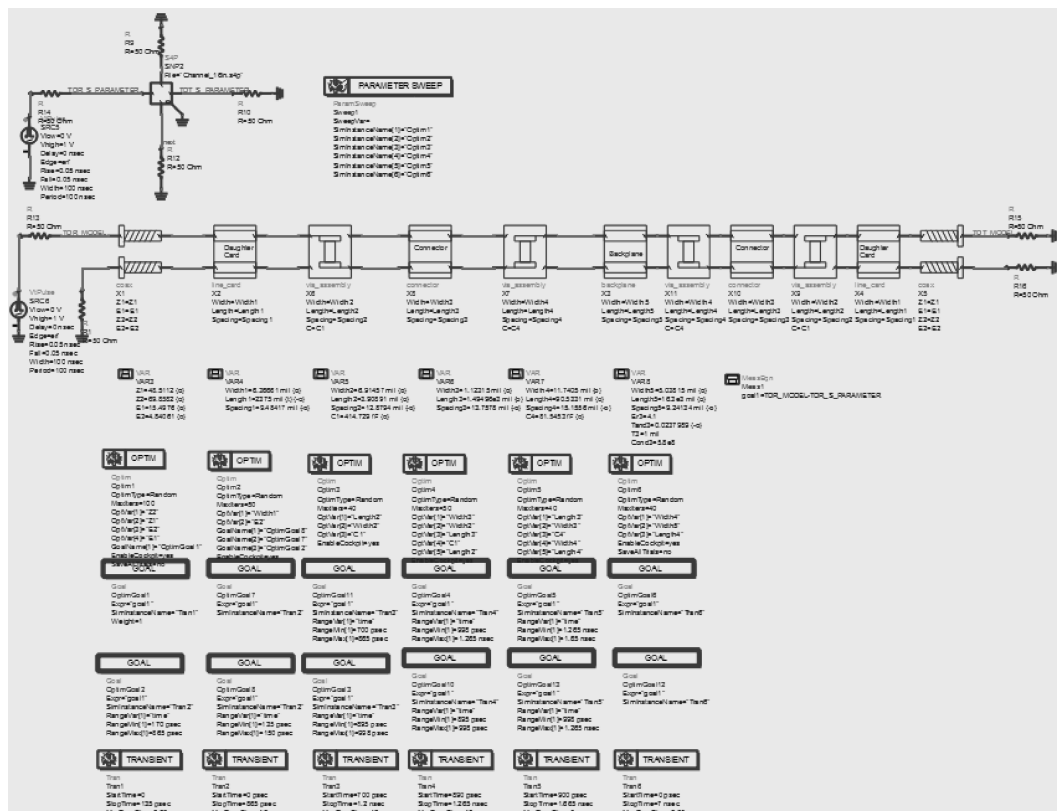


图 4-14 对等效电路模型进行优化

在原理图中可以看到加入了多个优化控制器,从上一节的电路参数调谐过程中也可以观察到,等效电路模型中位置靠近激励源的单元,其时域响应在时间上也靠前。原理图中的多个优化控制器就是利用这一特点,对不同的物理通道单元进行各自的优化。在 ADS 软件中,使用 Parameter Sweep 控制器可以进行按照指定顺序进行优化,图 4-15 给出了在 ADS 软件中进行优化的流程。

图 4-14 中仿真控制器的最顶层使用了扫描控制器,可以输入进行顺序仿真的控制器名称,在本实例中分别为 Optim1 到 Optim6;仿真开始时首先调入①,即 Optim1;在 Optim1 优化控制器中分别设定了优化类型(本实例中为 Random——随机优化)、优化最大迭代次数(本例中为 50)、可优化的变量及范围(本实例中有四个优化变量,分别制定了变量取值范围。注:在 ADS 软件原理图菜单 Options→Preferences...→Component Text/Wire Label 中可以设定优化变量在原理图中的显示方式:完整显示、缩略显示及不显示)以及优化目标 OptimGoal1;优化目标 OptimGoal1 中设定了优化目标表达式、得到优化目标表达式所需要运行

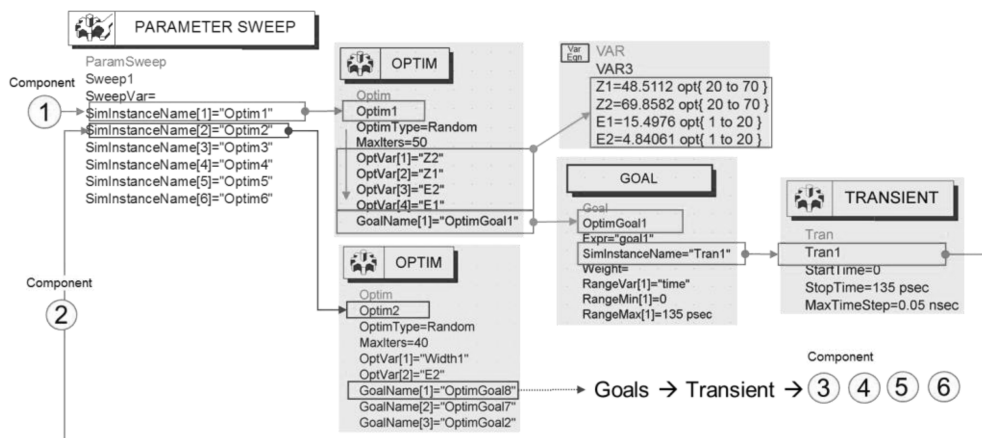


图 4-15 等效电路模型优化过程及要素

的仿真控制器 Trans1 以及计算优化目标的变量取值范围；最后是仿真控制器 Trans1，其中设定了最基本的瞬态仿真参数如仿真开始及结束时间和仿真步长。在完成第一轮仿真优化之后，再按照顺序进行②、③、④、⑤和⑥项目的优化，从而实现对整个通道的优化。

图 4-16 给出了针对第一节 SMA 接头的优化设置，有 4 个可优化变量，分别是接头模型中两段传输线的阻抗和电长度，优化控制器中使用随机优化方法对这 4 个变量进行优化，优化目标的名称是 OptimGoal1，在优化目标 OptimGoal1 中优化目标表达是 goal1，并使用瞬态仿真控制器 Tran1 进行分析。MeasEqn 中给出了优化目标表达式，是通道模型输入端的节点电压和测量模型输入端的节点电压之差。同时在优化目标 OptimGoal1 中指定优化目标表达式的计算范围是从 0 ~ 135psec 之间。

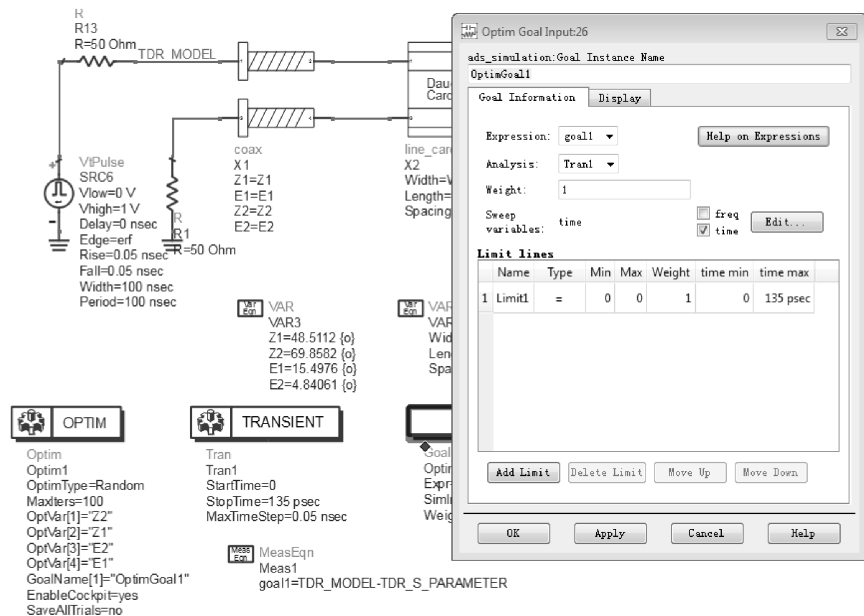


图 4-16 优化目标的设置

同样，在图 4-17 中的 Optim2 优化控制器中对第一节接头中的第二段传输线电长度以及

第二节中的传输线模型的宽度进行优化，和上面有所不同的是有三个优化目标，这三个优化目标都是使用瞬态仿真得到的（仿真控制器中仿真的时间范围是 0 ~ 865psec），所不同的是这三个优化目标中时间变量的取值范围不同，OptimGoal2 时间范围是 170 ~ 865psec，OptimGoal7 时间范围是 0 ~ 135psec，OptimGoal8 时间范围是 135 ~ 150psec。同时通过对这三个优化目标设定不同的权重（weight，OptimGoal7 设定为 10，OptimGoal8 设定为 4，OptimGoal2 设定为 1），从而使优化更偏向某一个目标。

仿真控制器仿真时间选择如图 4-18 所示。

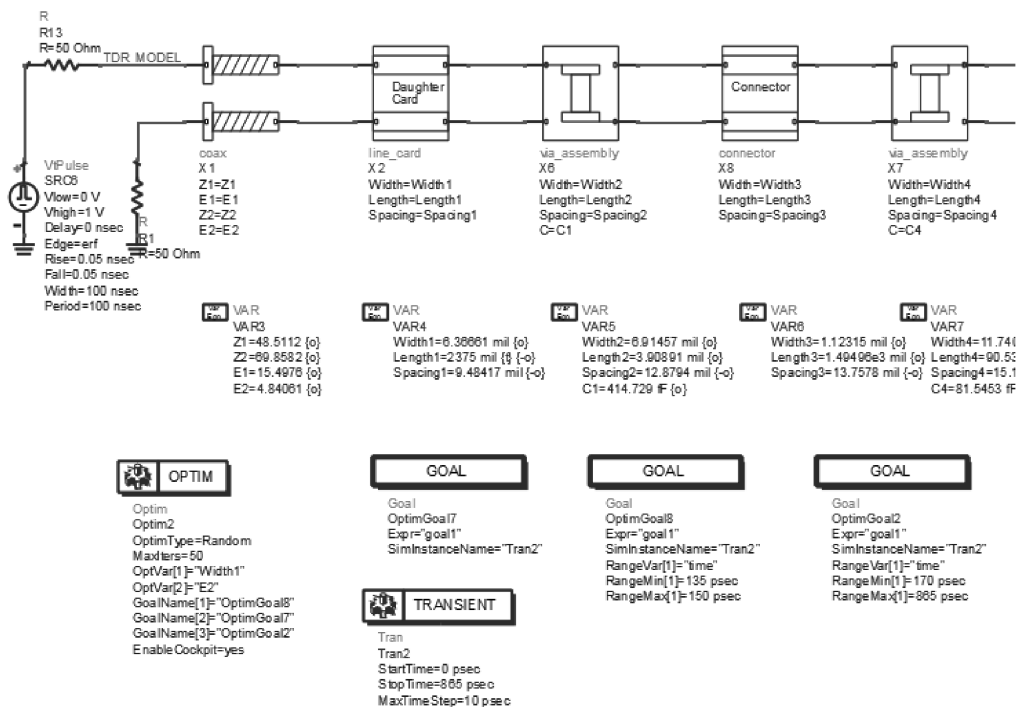


图 4-17 优化目标、优化控制器、优化变量及仿真控制器

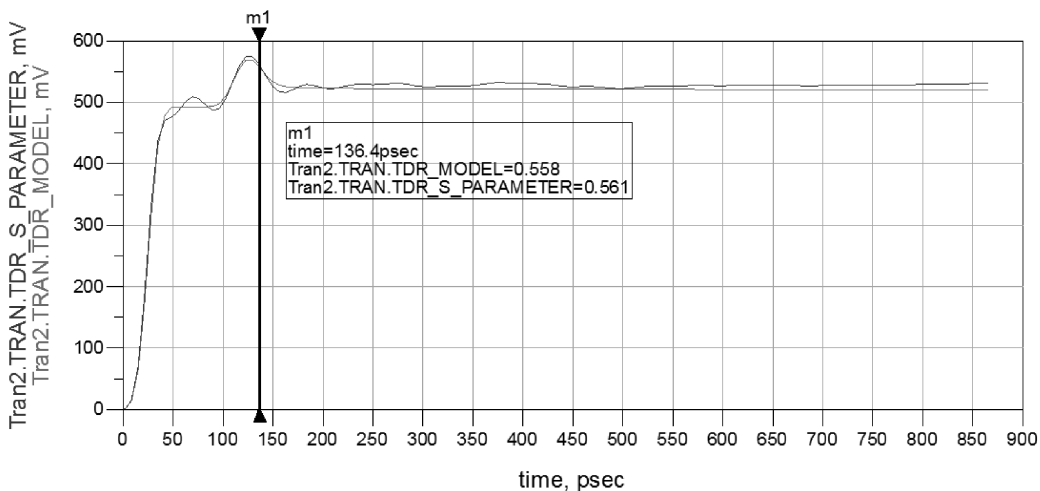


图 4-18 仿真控制器仿真时间选择

优化控制器 Optim3 以过孔单元的参数（线宽、线长及电容值）为优化变量，在 700psec 到 1.2nsec 之间进行瞬态仿真，分别设定 OptimGoal3（时间范围为 895psec 到 998psec，权重为 10）和 OptimGoal11（时间范围为 700psec 到 865psec，权重为 1）进行优化。在图 4-19 的数据显示图中，OptimGoal11 的时间范围是 m2 竖线标记以左的区域。竖线标记 m3 和 m4 之间的区域是 OptimGoal3 的优化目标时间范围。

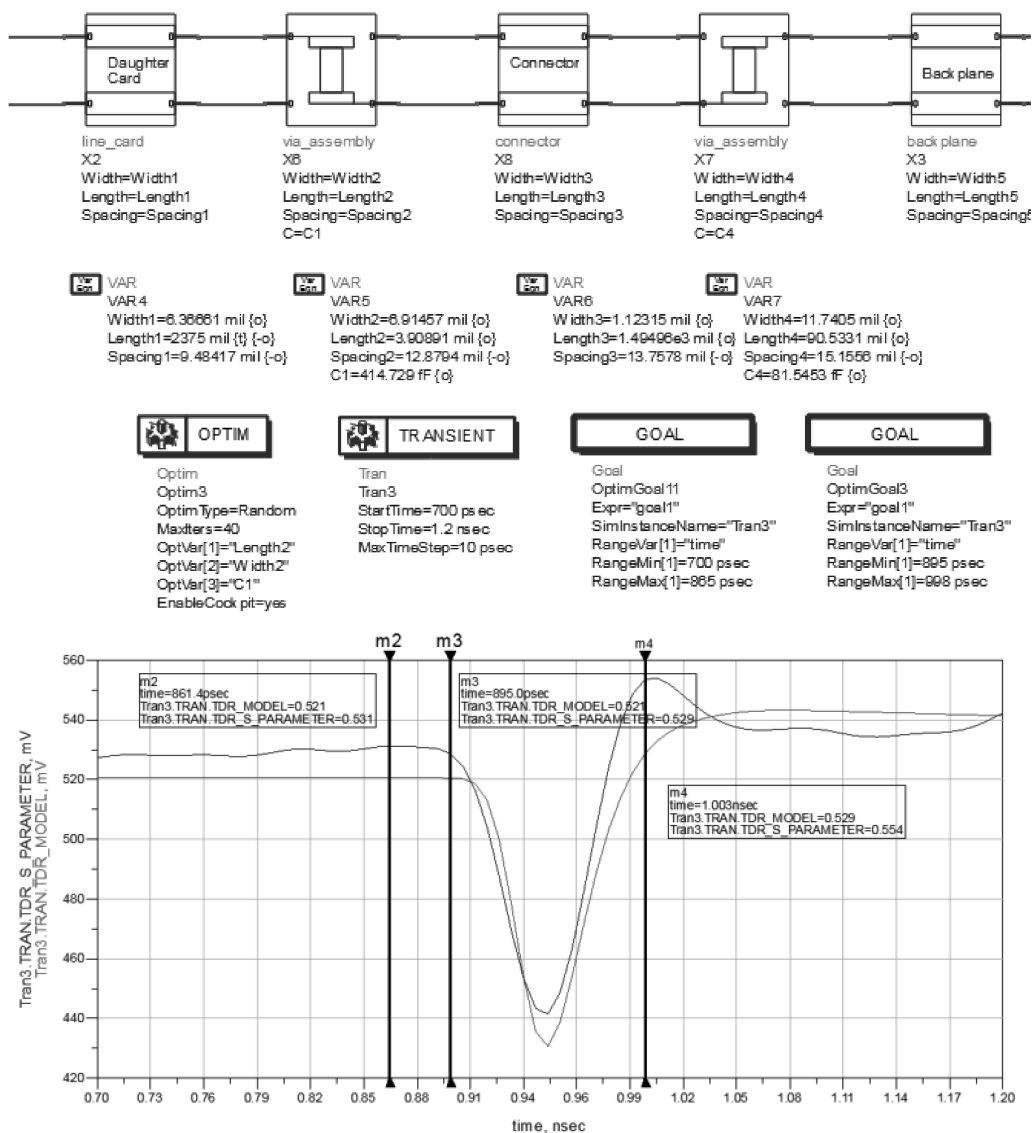


图 4-19 Optim3 优化设定及范围选择

单击原理图工具栏快捷区中的 按钮，就可以开始运行优化。优化开始后，会弹出优化座舱窗口，显示优化的各种信息以及优化的过程，如图 4-20 所示。

优化完成后，在 Tran6 的结果中显示出整个通道的优化结果及通道测量结果，如图 4-21 所示。

关闭优化座舱窗口，单击更新原理图中的变量，完成优化过程。



图 4-20 ADS 软件中的优化控制面板

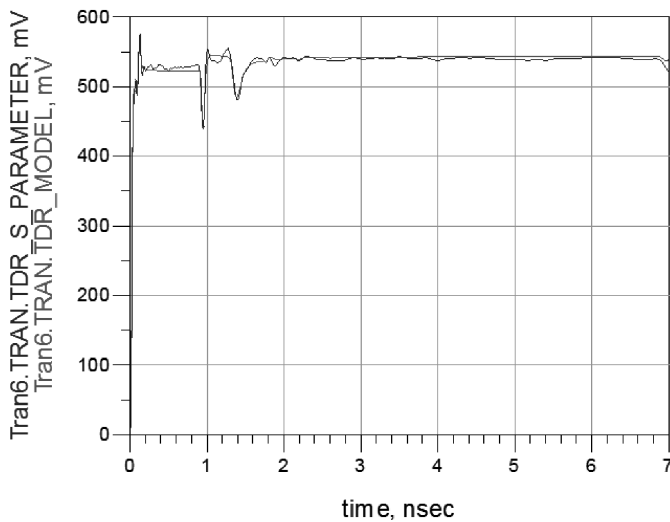


图 4-21 等效电路优化结果（虚线）及测量结果（实线）对比



## 4.5 使用优化通道模型进行仿真

在得到通道模型后就可以使用该模型进行更多的仿真与调试，如进行通道仿真，图 4-22 为使用通道模型运行 2.5G 速率的通道仿真，可显示眼图结果。

在此通道模型基础之上，还可以做更多的调试工作，如尝试修改板材的材料及层叠参数，进行“如果进行这样的改进—结果会是怎样的”验证，进一步优化通道的设计。

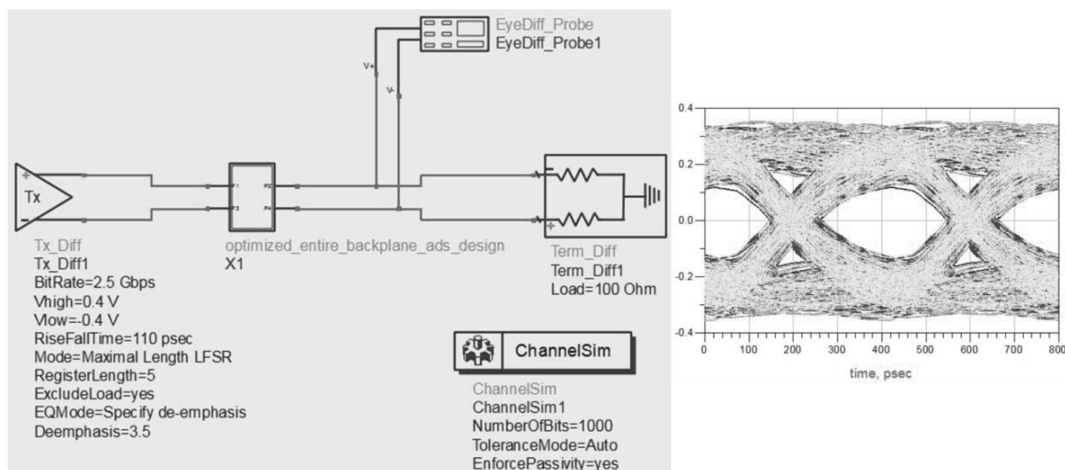


图 4-22 使用优化后的等效电路模型进行通道仿真

### 【总结】

本实例中根据测量得到的物理通道参数，使用电路建模的方法建立通道的电路模型。使用 ADS 软件的优化功能对电路参数进行优化，从而实现电路模型仿真结果和物理通道的测试结果一致。得到电路模型之后，可以进行更多的仿真和验证，帮助进行物理通道的优化。

# 实例 5 PCB 板材宽带参数提取

## 【目的】

本实例主要展示如何使用 ADS 处理网络分析仪测量的传输线的 S 参数，从测量的数据中提取板材介质参数，提高设计精度。实例内容包括编写 AEL 函数、后处理验证以及去嵌入等内容。

通过本实例学习，用户将涉及的操作方法：

- ☺ 使用 AEL 语言编写 Equation，进行后处理；
- ☺ 去嵌入模块的使用方法；
- ☺ S 参数文件的使用和仿真。

## 【背景知识】

### 1. 去嵌入 (De-embedding) 技术

在实际的测量过程中，测试的结果包含了测试夹具所带来的误差。我们可以把被测件和夹具描述成一组 S 参数，如图 5-1 所示。

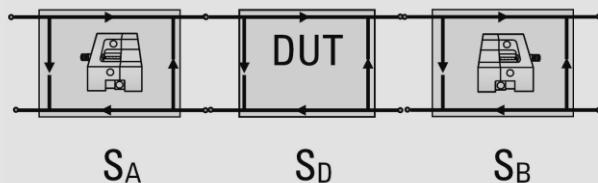


图 5-1 通过夹具测量被测件

☺  $S_D$ ：代表被测件的 S 参数；

☺  $S_A/S_B$ ：代表被测件左右两侧夹具的 S 参数。

去嵌入技术就是在已知夹具的 S 参数 ( $S_A/S_B$ ) 的基础上，把被测件的 S 参数从总的测量的被测件和夹具一起的 S 参数中分离出来的运算过程。

去嵌入技术运用了电路网络的矩阵计算。为了运算上的方便往往先将 S 参数转换到 T 参数来进行级联运算。

$$[T] = \begin{bmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{bmatrix}$$

所以对于夹具上的器件最终测量的结果为

$$[T_{\text{MEASURED}}] = [T_A][T_{\text{DUT}}][T_B]$$

因此只要事先获得测试夹具的 T 矩阵就可以在测试中将其的影响消除掉，具体公式如下：



$$[T_A]^{-1}[T_A][T_{DUT}][T_B][T_B]^{-1} = [T_{DUT}]$$

一般的去嵌入步骤为：

(1) 获得测试夹具的数学模型，可以使用 S 参数或者 T 参数来分别表示测试夹具的每半个部分；

(2) 对矢量网络分析仪做全双端口校准以及测量，测量结果包括夹具以及被测件的数据；

(3) 将 S 参数转换到 T 参数；

(4) 使用去嵌入的公式对测量结果进行去嵌入，即

$$[T_{DUT}] = [T_A]^{-1}[T_M][T_B]^{-1}$$

(5) 将结果转换为 S 参数  $S_D$ 。

## 2. AFR 自动夹具移除校准技术

自动夹具移除 (AFR) 校准技术是一种提取准确的宽带夹具模型的简便方法。这种校准技术可以被用于各种夹具和互连的结构，如转接头、芯片封装、线缆、PCB 印刷传输线以及通孔这样的互连结构。这种校准技术和传统的 TRL 校准技术拥有同样的高精度校准性能，却有着更为简单的夹具制作实现。AFR 在高速信号完整性领域有着广泛的应用。

图 5-2 展示了一块通孔作为被测件的测试板。通孔结构作为被研究的对象处于两段均匀传输线的中间，传输线的两端是 SMA 的转接器，用来连接网络分析仪，测量通孔的 S 参数。在这个例子中，我们关心的被测件是通孔。为了进行测量，通孔处于夹具的中间 (夹具包括 SMA 连接器以及连接通孔的传输线)。从图 5-2 上蓝色的 TDR 响应曲线可以看出，SMA 转接器带来了不可忽视的不连续性，传输线也不是完全均匀的。在传输过程中可以观测到阻抗的波动，同时也带来了传输损耗。

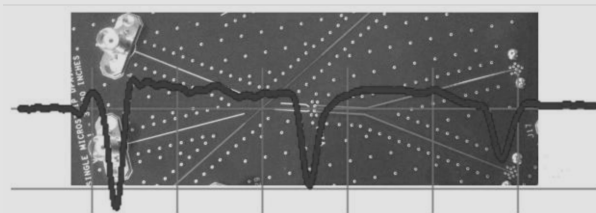


图 5-2 测试板

所有的测试测量所面临的一个共同的基础问题正如以上的例子，即如何把被测件的测量结果从整个测量结果 (被测件加上夹具) 中分离出来。这也是 AFR 自动夹具移除校准技术所要解决的问题。

目前业内已经有很多测试测量的校准技术，包括 SOLT (短路、开路、负载、直通)、TRL (直通、反射、传输线) 和 LRM (传输线、反射、匹配)，这三种校准技术在实现的复杂程度和提取被测件测量结果的准确性上做了折中。不幸的是，对于被广泛在测试测量中用于移除夹具影响的 TRL 和 LRM 校准技术，在实现过程中，对于夹具正确制作有很高的要求，较容易引入人工误差。对于校准精度的要求越高，夹具的实现就越困难。

而自动夹具移除 (AFR) 校准技术在校准精度和实现的容易程度这对原本的矛盾的关系上取得了突破, 在具有极高校准精度的同时夹具的设计也非常简单。

AFR 技术是在被测件两边的夹具镜像对称的情况下实现的。在这样的情况下, 需要做一个夹具的校准件来提取夹具的 S 参数。校准件的形式是把两侧的夹具直接连接在一起形成一个两倍于单侧夹具长度的直通结构。这种校准件通常被叫做 2X 直通参考夹具, 如图 5-3 所示。

虽然单侧的夹具并不是对称的, 但当两个对称的夹具级联后, 新的 2X 直通参考夹具校准件是镜像对称的, 所以通过测试得到的校准件的 S 参数中,  $S_{11} = S_{22}$ ,  $S_{21} = S_{12}$ , 可以得到两个已知量, 并不足够求解出单侧夹具 S 参数 ( $S_{21_A} = S_{12_A}$ ) 的三个未知量, 如图 5-4 所示。

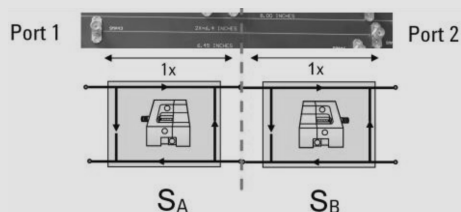


图 5-3 2X 直通参考夹具

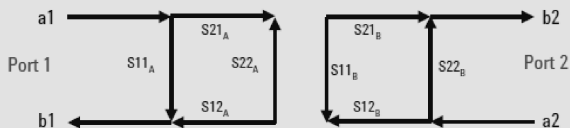


图 5-4 S 参数级联

而 AFR 技术基于 2X 直通参考夹具校准件的中间包含一段均匀的传输线这一特性, 通过采用时域信号处理的方法可以提取出夹具的  $S_{11_A}$  和  $S_{22_B}$ 。借助多出来的一个已知量, 单侧夹具的 S 参数就可以被唯一求解出。利用去嵌入技术, 夹具的影响就可以从测试结果中去除, 得到被测件的 S 参数。

### 3. Svensson/Djordjevic 介质模型

介质的介电常数, 又称电容率, 是电位移  $D$  与电场强度  $E$  之比。而有耗介质的损耗通常用复介电常数虚部的损耗角正切 ( $\tan D$ ) 来表示。

$$\epsilon = \epsilon_r - j \cdot (\epsilon_r \cdot \tan D)$$

为了保证介质的因果性, ADS 中使用了 Svensson/Djordjevic 模型作为有耗介质的宽带模型

$$\epsilon(\text{freq}) = \epsilon_{\infty} + a \cdot \ln \frac{f_H + j \cdot \text{freq}}{f_L + j \cdot \text{freq}}$$

其中  $f_L$  和  $f_H$  是模型的参数

$$f_L = \text{LowFreqForTanD}$$

$$f_H = \text{HighFreqForTanD}$$

$\epsilon_{\infty}$  为当频率接近无穷大时介电常数的值,  $a$  是一个常数。 $\epsilon_{\infty}$  和  $a$  这两个参数可以从 ADS 中用户输入的其他参数 ( $\epsilon_r/\tan D/\text{FreqForEpsrTanD}/\text{LowFreqForTanD}/\text{HighFreqForTanD}$ ) 中计算出来。

图 5-5 和图 5-6 显示的是采用 Svensson/Djordjevic 模型的介电常数随频率变化的趋势, 介质的具体参数如下:



图 5-5 介电常数的实部随频率的变化

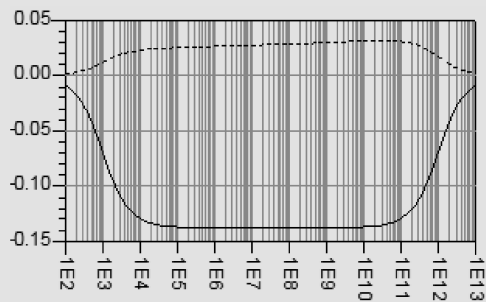


图 5-6 介电常数的虚部以及损耗角正切随频率的变化

- ☺  $E_r = 4.6$
- ☺  $TanD = 0.03$
- ☺  $FreqForEpsrTanD = 1\text{ GHz}$
- ☺  $HighFreqForTanD = 1\text{ THz}$
- ☺  $LowFreqForTanD = 1\text{ kHz}$



## 5.1 测量文件的导入和验证

### 1. 模型的准备

用来提取 PCB 介电常数的电路是一根长度  $L$  为 1inch 的带状线，带状线两侧分别是夹具 A 和夹具 B，夹具由 SMA 转接器、焊盘和传输线组成，夹具 A 和夹具 B 设计成镜像对称的结构，如图 5-7 所示。

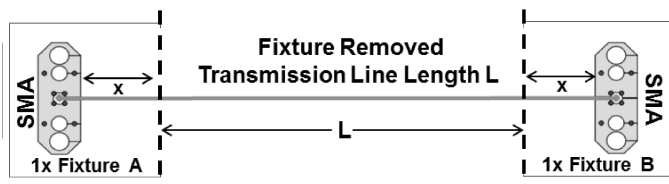


图 5-7 待测电路

使用 Keysight 的网络分析仪测试图 5-7 的电路的  $S$  参数，端口阻抗是 50ohm，测量频率的范围是 10MHz ~ 40GHz。把测量的结果保存为 touchstone 格式的文件 MD\_2p4in\_T.s2p。

为了得到长度  $L$  为 1inch 的带状线的  $S$  参数，需要移除夹具对  $S$  参数的影响。这里选择采用前面介绍的 AFR 技术，需要测量夹具 A 和夹具 B 相连接的 2X 直通参考夹具的  $S$  参数。

设计的 2X 直通参考夹具如图 5-8 所示，夹具 A 和夹具 B 是镜像对称的。但是由于电路的制作存在工艺上的不稳定和误差，夹具 A 和夹具 B 的  $S$  参数以及图 5-7 和图 5-8 中每个夹具的  $S$  参数，存在着一定幅度的不一致。

使用 Keysight 的网络分析仪测试图 5-8 的电路的  $S$  参数，端口阻抗是 50ohm，测量频率

的范围是 10MHz ~ 40GHz。把测量的结果保存为 touchstone 格式的文件 MD\_1p4in\_T.s2p。

打开 Keysight ADS 软件，新建工程文件 PCB\_Parm\_Extraction\_wrk，假设工程文件所在目录为 \PCB\_Parm\_Extraction\_wrk。

将实例下载资源压缩文件中的 .s2p 文件（见图 5-9）放入 \PCB\_Parm\_Extraction\_wrk \data 文件夹中，有四个文件。

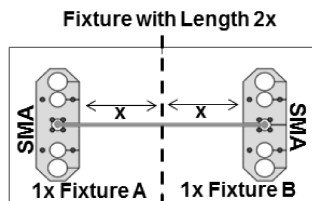


图 5-8 2X 直通参考夹具

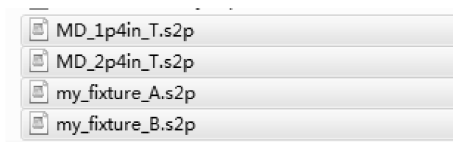


图 5-9 测量的 S 参数文件

## 2. S 参数的验证

新建原理图 al\_meas\_data，按照图 5-10 所示编辑原理图，并编辑下列模块和仿真器的参数。

S\_Param:

Start = 0GHz  
Stop = 40GHz  
Step = 0.01GHz  
EnforcePassivity = yes

S2P

File = "MD\_1p4in\_T.s2p"  
Type = Touchstone

S2P

File = "MD\_1p4in\_T.s2p"  
Type = Touchstone

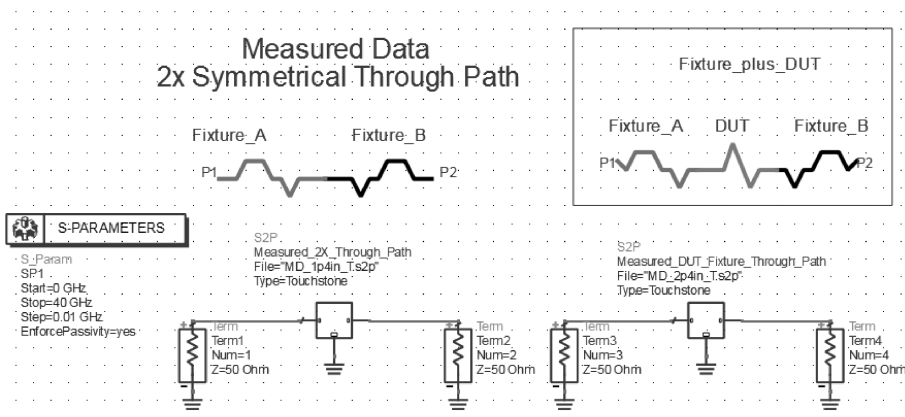


图 5-10 原理图 al\_meas\_data

单击快捷图标“Simulate”运行仿真，在跳出的显示窗口中使用左侧的快捷图标“Rectangular Plot”显示 2X 直通参考夹具电路和被测件 + 夹具电路的 S 参数（见图 5-11）。

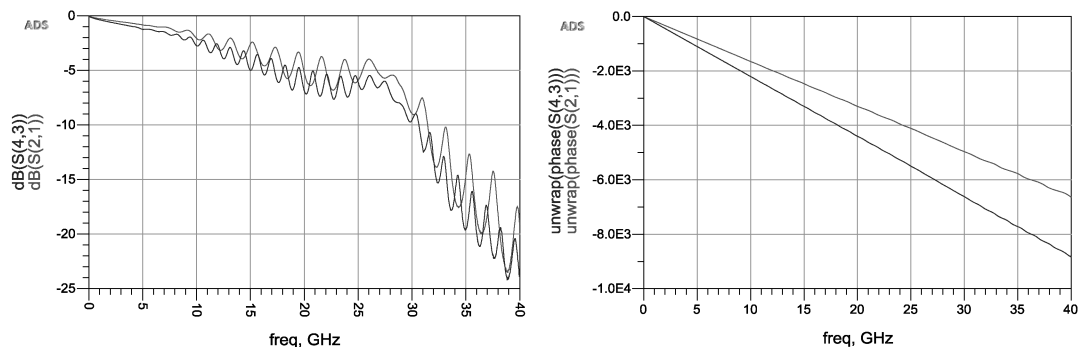


图 5-11 S 参数仿真结果

- ☺  $\text{dB}(S(2,1))$
- ☺  $\text{dB}(S(4,3))$
- ☺  $\text{unwrap}(\text{phase}(S(2,1)))$
- ☺  $\text{unwrap}(\text{phase}(S(4,3)))$

图 5-11 的 S 参数与测试过程中网络分析仪的 S 参数完全一致，同时从相位曲线中可以看出，两组相位中相差了被测件带来的传播时延。还可以从时域曲线上验证以上结论，编辑 Equation Fixture\_2X\_TDR 和 Fixture\_plus\_DUT\_TDR（见图 5-12），通过“Rectangular Plot”显示时域 TDR 的曲线（见图 5-13），2X 直通参考夹具具有两个明显的不连续性（夹具 A 和夹具 B），同时加上 DUT 电路的时延要大于直通的夹具。

```
Eqn Fixture_2X_TDR=tdr_sp_imped(S(1,1),0.05ns,50,-0.2ns,2.2ns,401,4)
Eqn Fixture_plus_DUT_TDR=tdr_sp_imped(S(3,3),0.05ns,50,-0.2ns,2.2ns,401,
```

图 5-12 公式

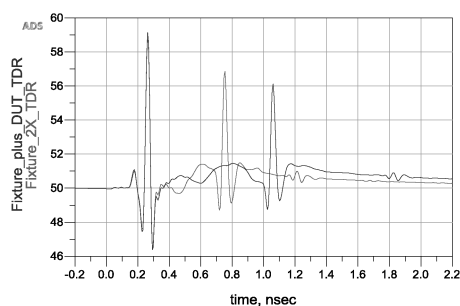


图 5-13 S 参数转 TDR 结果



## 5.2 AFR 和测量结果去嵌入

### 1. 夹具 S 参数的抽取

运行 Keysight Physical Layer Test System 软件（以下简称 PLTS），在主窗口运行菜单栏 Utilities→Automatic Fixture Removal→Wizard...（见图 5-14）。

在弹出的窗口中进行 AFR 每一步的设置，在页面 1，选择单端信号双端口测量（见图 5-15）。

在页面 2 中选择“2X Thru”（见图 5-16），即夹具 A 和夹具 B 直通，在 AFR 技术的双端口直通 S 参数提取中，认为夹具 A 的 S21 和夹具 B 的 S12 传输函数是一样的。

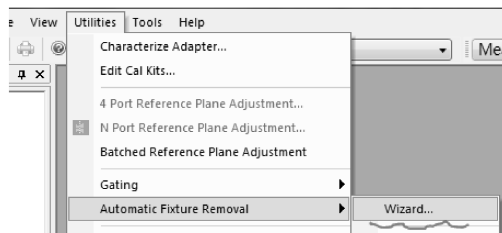


图 5-14 运行 AFR 选项

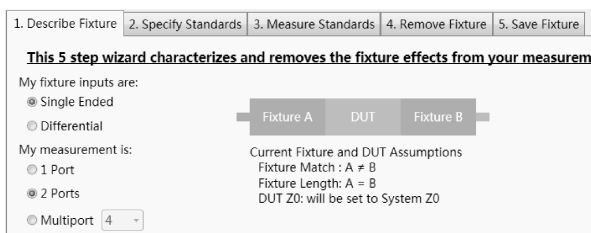


图 5-15 AFR - 页面 1

在 AFR wizard 页面 3 中 load 之前测量得到的 S 参数文件“MD\_1p4in\_T.s2p”。Load 之后会在页面的下方计算出夹具中传输线的特性阻抗以及夹具的传输时延，如图 5-17 所示。

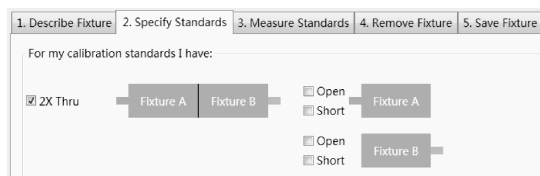


图 5-16 AFR - 页面 2



图 5-17 AFR - 页面 3

在页面 4 中选择“De-embedding for PLTS measurements”（见图 5-18）。

在页面 5 中按照图 5-19 进行选择，并选择输出文件名“my\_fixture”，单击“Save Fixture Files”，将会有两个 .s2p 文件生成，分别代表两个夹具的 S 参数。

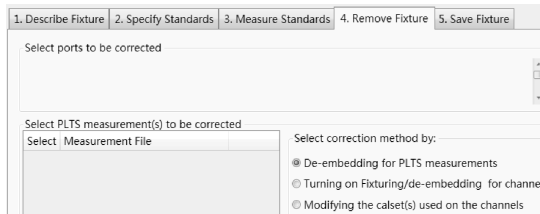


图 5-18 AFR - 页面 4

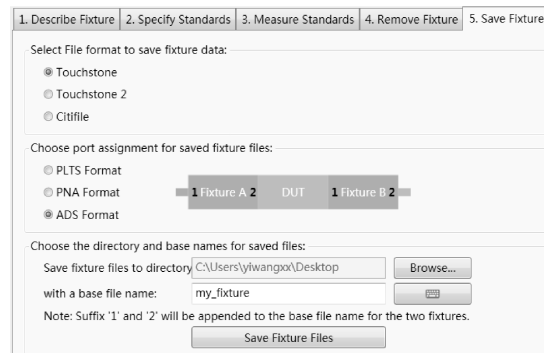


图 5-19 AFR - 页面 5

## 2. 夹具 S 参数的验证

新建原理图 b1\_fixture\_PLTS，按照图 5-20 编辑原理图，并编辑下列模块和仿真器的参数。其中使用快捷图标中的“Mirror about Y axis”对红圈标注的 S2P 模块进行操作。

S\_Param:

Start = 0GHz

Stop = 40GHz

Step = 0.01GHz

S2P

File = "my\_fixture\_A.s2p"  
 Type = Touchstone  
 S2P  
 File = "my\_fixture\_B.s2p"  
 Type = Touchstone

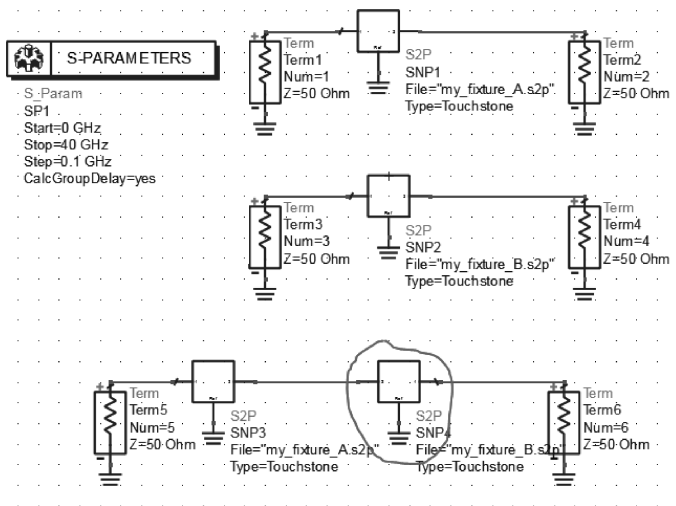


图 5-20 原理图 b1\_fixture\_PLTS

运行仿真，在显示窗口通过“Rectangular Plot”和“Smith”显示原理图中端口 1 和端口 2 所有的 S 参数，如图 5-21 所示，即夹具 A 的 S 参数。

### 1) 添加 Equation

☺ AB\_dataset = "a1\_meas\_data"

☺ PLTS\_dataset = "b1\_fixture\_PLTS"

### 2) 添加“Rectangular Plot”

☺ dB(\$AB\_dataset . S(1,1)) 和 dB(\$PLTS\_dataset . S(5,5))

☺ phase(\$AB\_dataset . S(1,1)) 和 phase(\$PLTS\_dataset . S(5,5))

☺ dB(\$AB\_dataset . S(2,2)) 和 dB(\$PLTS\_dataset . S(6,6))

☺ phase(\$AB\_dataset . S(2,2)) 和 phase(\$PLTS\_dataset . S(6,6))

☺ dB(\$AB\_dataset . S(2,1)) 和 dB(\$PLTS\_dataset . S(6,5))

☺ phase(\$AB\_dataset . S(2,1)) 和 phase(\$PLTS\_dataset . S(6,5))

☺ dB(\$AB\_dataset . S(1,2)) 和 dB(\$PLTS\_dataset . S(5,6))

☺ phase(\$AB\_dataset . S(1,2)) 和 phase(\$PLTS\_dataset . S(5,6))

### 3) 添加 Smith 圆图

☺ \$AB\_dataset . S(1,1) 和 \$PLTS\_dataset . S(5,5)

☺ \$AB\_dataset . S(2,2) 和 \$PLTS\_dataset . S(6,6)

☺ \$AB\_dataset . S(1,2) 和 \$PLTS\_dataset . S(5,6)

☺ \$AB\_dataset . S(2,1) 和 \$PLTS\_dataset . S(6,5)

从图 5-22 可以看出夹具 A 和夹具 B 级联的参数可以恢复到拆解之前的参考校准件。

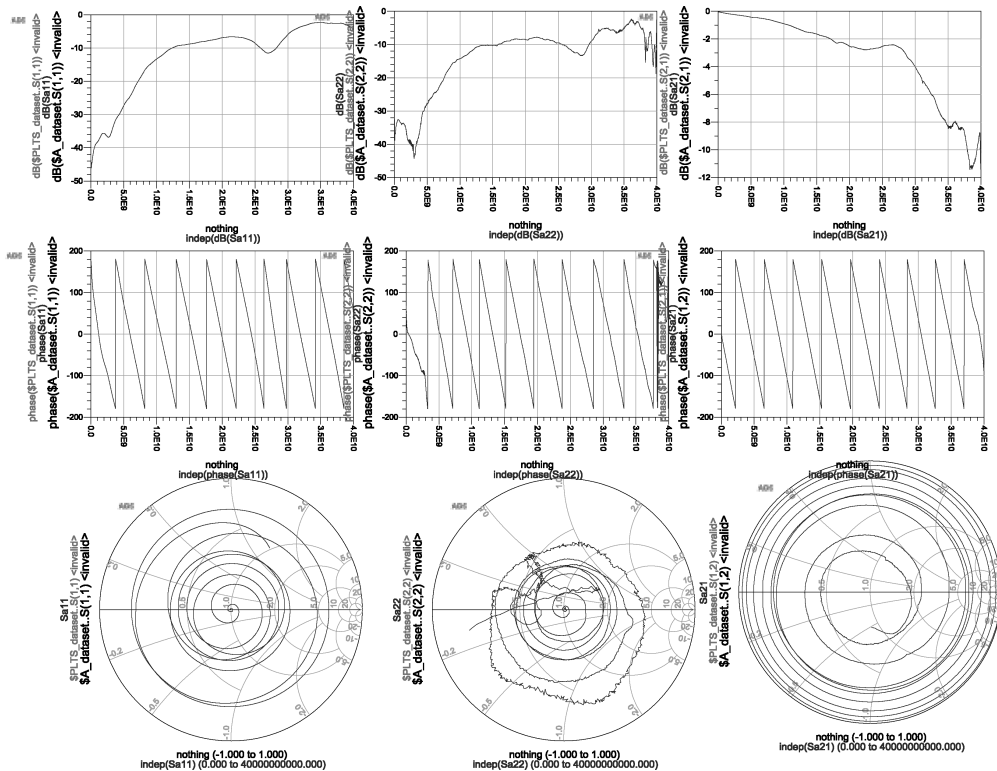


图 5-21 S 参数仿真结果

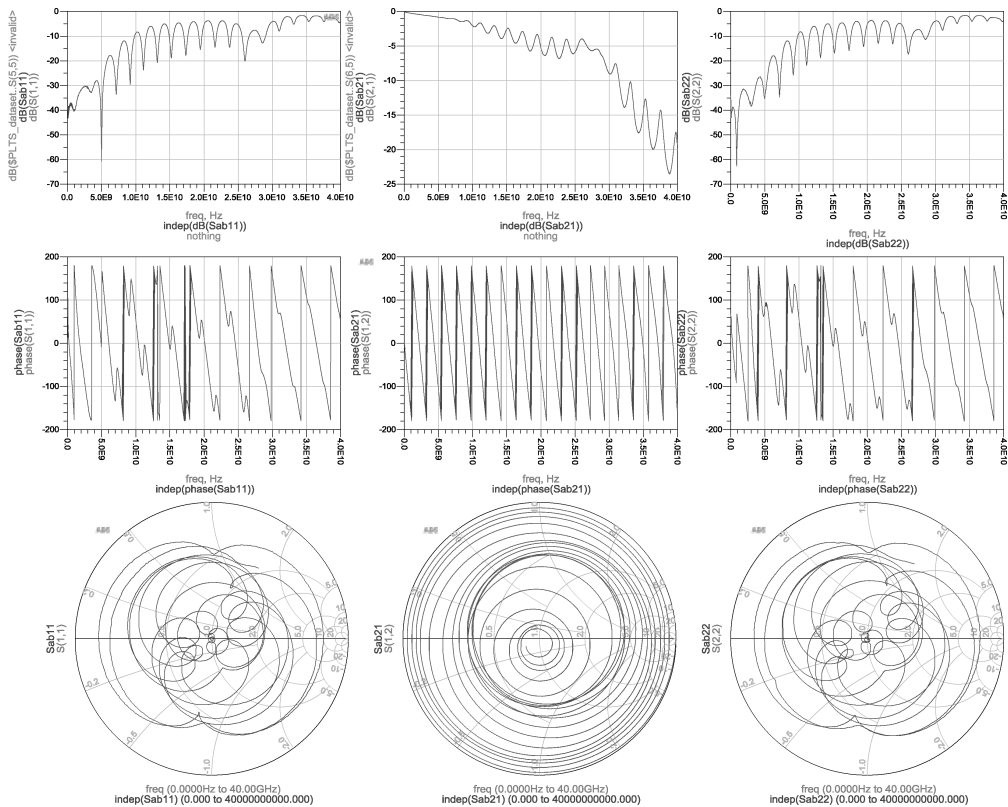


图 5-22 S 参数仿真结果



在图 5-13 的显示页面, 添加 Equation, 并在图 5-13 上添加 “Fixture\_A\_TDR”。夹具 A 的 TDR 的不连续性和之前被测件上左侧的夹具响应一致, 如图 5-23 所示。

$\text{Fixture\_A\_TDR} = \text{tdr\_sp\_imped}(\text{Sa11}[0, :, :], 0.05\text{ns}, 50, -0.2\text{ns}, 2.2\text{ns}, 401, 4)$

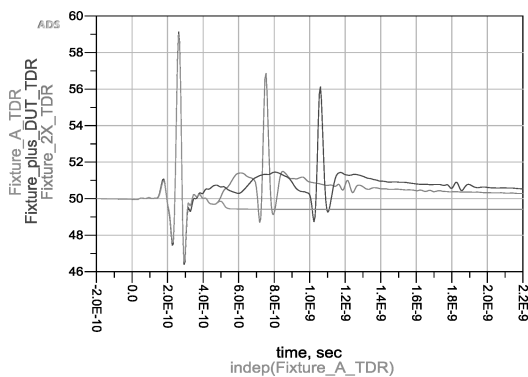


图 5-23 TDR 结果

### 3. 去嵌入

新建原理图 c1\_TML\_deembed, 按照图 5-24 编辑原理图, 并编辑下列模块和仿真器的参数。

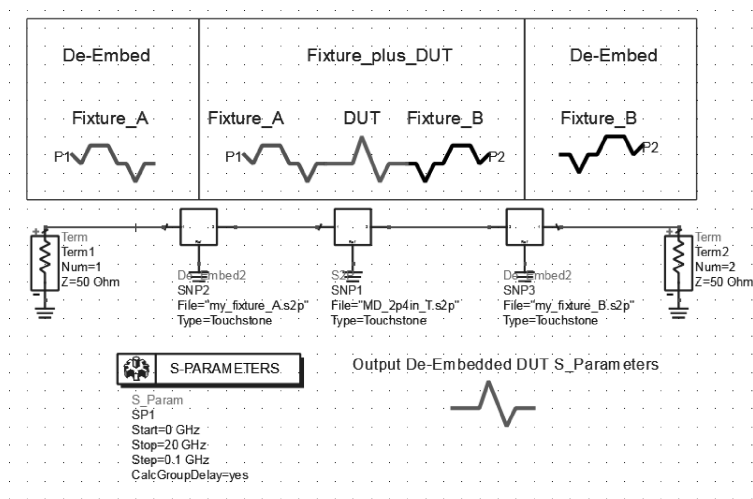


图 5-24 原理图 c1\_TML\_deembed

S\_Param:

Start = 0GHz

Stop = 20GHz

Step = 0.01GHz

CalcGroupDelay = yes

左 De\_Embed2

File = " my\_fixture\_A. s2p"

Type = Touchstone

S2P

File = "MD\_2p4in\_T. s2p"

Type = Touchstone

右 De\_Embed2

File = "my\_fixture\_B. s2p"

Type = Touchstone

运行仿真，仿真的结果是得到 1in 带状线的 S 参数（见图 5-25）。

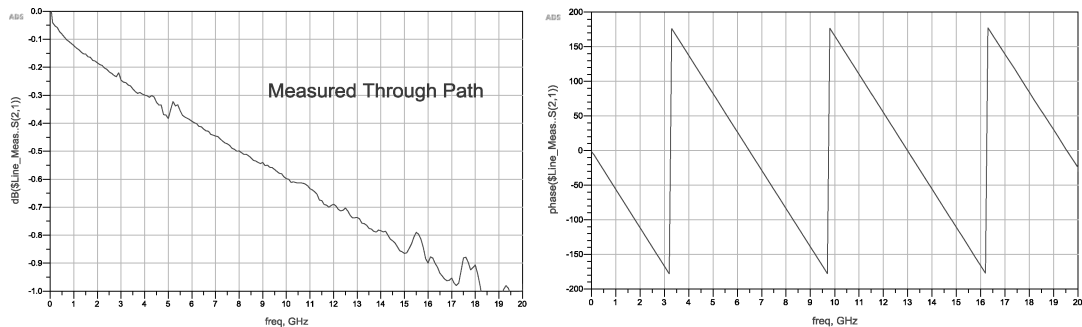


图 5-25 S 参数仿真结果



## 5.3 介电常数拟合

### 1. 介电常数拟合

新建原理图 d1\_Material\_Properties\_from\_Test\_Structures，按照图 5-26 编辑原理图，并编辑下列模块和仿真器的参数。

S\_Param:

Start = 0GHz

Stop = 20GHz

Step = 0.01GHz

CalcGroupDelay = yes

VAR

Er = 3.2 tune { 2 to 5 by 0.01 }

TanD = 0.006 tune { 0.001 to 0.03 by 0.001 }

t = 0.6

sigma = 4e7

MLSUBSTRATE3

Er[1] = Er

H[1] = 8.5 mil

TanD[1] = TanD

T[1] = t mil

Cond[1] = sigma

```

Er[2] = Er
H[2] = 8.5 mil
TanD[2] = TanD
T[2] = t mil
Cond[2] = sigma
T[3] = t mil
Cond[3] = sigma
LayerType[1] = ground
LayerType[2] = signal
LayerType[3] = ground
DielectricLossModel = Svensson/Djordjevic
FreqForEpsrTanD = 10GHz
Rough = 0 mil

```

ML1CTL\_C

```

Subst = "Subst1"
Length = 1000 mil
W = 9 mil
Layer = 2

```

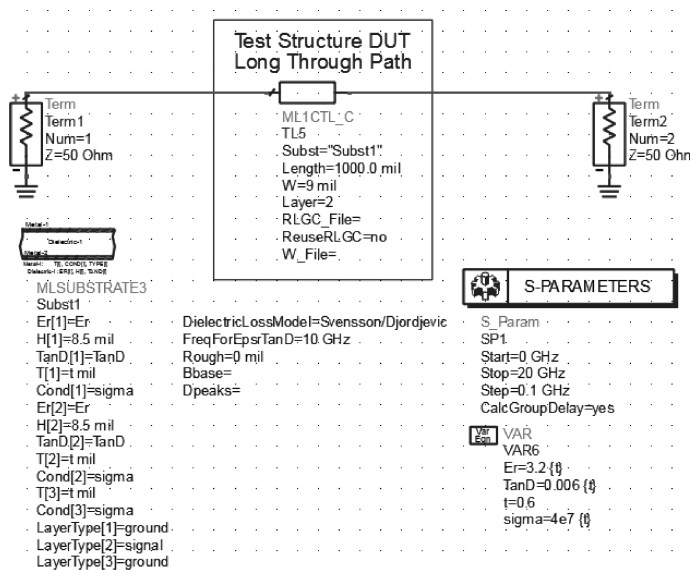


图 5-26 原理图 d1\_Material\_Properties\_from\_Test\_Structures

单击快捷图标（见图 5-27）中的“tuning”，运行调试。



图 5-27 快捷图标

在显示的窗口使用“Equation”编辑公式，并使用“Rectangular Plot”画出多条曲线。

Equation:Line\_Meas = "c1\_TML\_deembed"

```

Plot1: phase(S(2,1)) - phase($Line_Meas. . S(2,1))
Plot2: phase($Line_Meas. . S(2,1)) - phase(S(2,1))
Plot3: dB(S(2,1)) - dB($Line_Meas. . S(2,1))
Plot4: dB($Line_Meas. . S(2,1)) - dB(S(2,1))

```

在弹出的窗口调整  $\epsilon_r$  和  $\tan\delta$ ，使得相位尽可能吻合，同时兼顾插损的一致。最终通过调试，在 10GHz 的频率上， $\epsilon_r = 3.285$ ， $\tan\delta = 0.007$ ，这一结果和 PCB 介质的 datasheet 一致。经过介电常数调整后，显示的结果如图 5-28 所示。

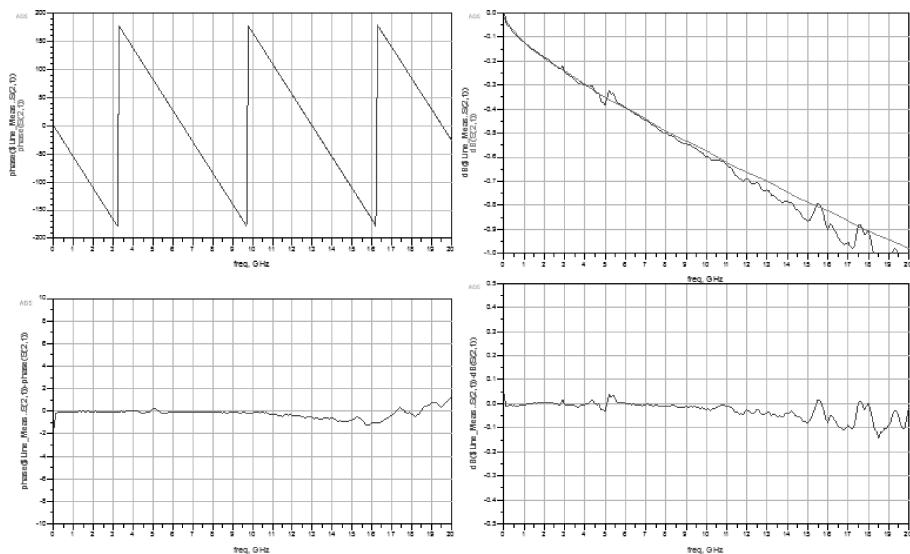


图 5-28 参数调整后仿真结果

## 2. 材料属性建模

新建数据显示窗口（见图 5-29）e1\_Material\_Properties\_dK\_df\_Svennsson\_Djordjevic。

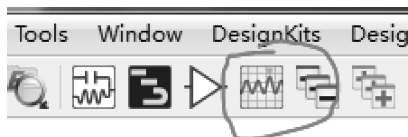


图 5-29 数据显示窗口快捷图标

按照图 5-30 编辑公式。

$\epsilon_r = 3.285$

$\tan\delta = 0.007$

$\tan\delta_{\text{Freq}} = 10\text{GHz}$

$\text{Low\_Freq} = 1\text{kHz}$

$\text{High\_Freq} = 1\text{THz}$

$\text{value\_}\tan\delta_{\text{Freq}} = \epsilon_r * (1 - j * \tan\delta)$

$\ln f = \ln((\text{High\_Freq} + j * \tan\delta_{\text{Freq}}) / (\text{Low\_Freq} + j * \tan\delta_{\text{Freq}}))$

$a = \text{imag}(\text{value\_}\tan\delta_{\text{Freq}}) / \text{imag}(\ln f)$

```
b = real( value_TanD_Freq ) - a * real( Inf)
powers = generate(2,13,111)
frequency = 10 * * powers
Er_vs_freq = b + a * ln( ( High_Freq + j * frequency)/( Low_Freq + j * frequency) )
TanD_vs_freq = - imag( Er_vs_freq)/real( Er_vs_freq)
```

**Input Values**

Eqn Er = 3.285	Eqn TanD_Freq= 10 GHz
Eqn TanD = 0.007	Eqn Low_Freq = 1 kHz
	Eqn High_Freq = 1 THz

**Equations**

```
Eqn value_TanD_Freq = Er*(1-j*TanD)
Eqn Inf = ln((High_Freq + j*TanD_Freq)/(Low_Freq + j*TanD_Freq))
Eqn a = imag(value_TanD_Freq) / imag(Inf)
Eqn b = real(value_TanD_Freq) - a*real(Inf)
Eqn powers=generate(2,13,111)
Eqn frequency=10**powers
Eqn Er_vs_freq = b + a * ln((High_Freq + j*frequency)/(Low_Freq + j*frequency))
Eqn TanD_vs_freq = -imag(Er_vs_freq)/real(Er_vs_freq)
```

图 5-30 公式

使用 “Rectangular Plot” 显示介电常数随频率的变化（见图 5-31）。

```
Trace 1:
Trace: plot_vs( real( Er_vs_freq) ,frequency)
Y with linear scale and X with log scale

Trace 2:
plot_vs( imag( Er_vs_freq) ,frequency)
plot_vs( TanD_vs_freq, frequency)
Y with linear scale and X with log scale
```

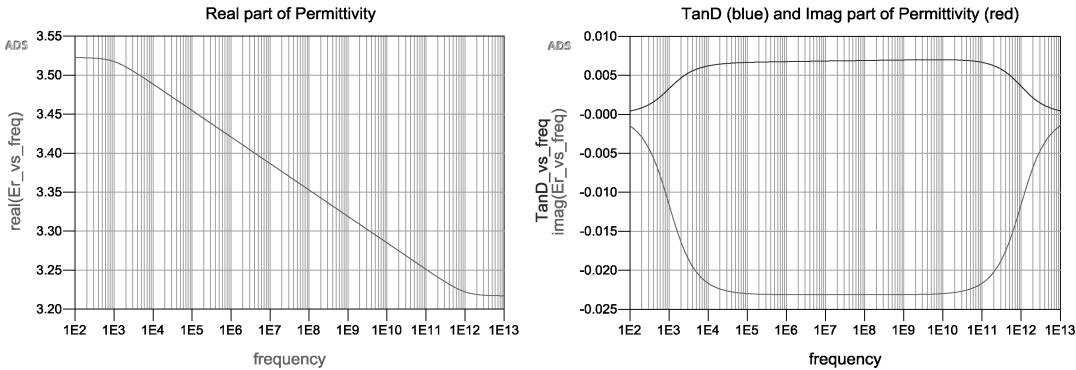


图 5-31 介电常数随频率的变化

【总结】

本实例主要展示了使用传输线来提取介质材料介电常数的方法。在流程中使用网络分析仪测量目标电路和参考夹具校准件的 S 参数，使用 PLTS 软件进行夹具移除，使用 ADS 软件完成去嵌入和最后数据的提取。通过仿真练习，可以学习 S 参数仿真及其后处理的方法，了解仿真与测试相互结合的流程，提升高速数字电路建模能力，提高针对高速信号完整性问题设计验证的能力。

## 实例 6 CEI - 25G - LR 通道特性 一致性分析

### 【目的】

本实例主要介绍如何将测量的 CEI - 25G - LR 通道数据根据协议定义的规范进行一致性分析。通过本实例，用户将学习批处理仿真方法，通过 AEL 脚本自定义函数及数据的后处理。本实例的测量方法对于其他高速数字接口，如 802.3ap, 802.3ba 等也有借鉴意义。

### 【背景知识】

Common Electrical I/O (CEI) 是 Optical Internetworking Forum 组织 (OIF) 提出的针对 6Gbps、11G + bps 及 25G + bps 互联网应用的通用电接口协议。协议定义了高速背板、芯片与芯片及光模块之间互联时的发射机、接收机及互联通道应当遵循的规范。协议自 2003 年开始起草，2014 年 2 月发布了 3.1 版本。新的版本主要增加了 28G 极短距离 (Very Short Reach, VSR) 和中距离 (Medium Reach, MR) 接口的规范。

本实例主要针对协议中 25G 长距离接口 (Long Reach Interface, LR) 部分，即 3.1 版本协议中第 11 章 (CEI - 25G - LR Long Reach Interface) 第 2 条第 6 款所定义的一致性测试 (channel compliance) 内容进行了实现。

典型的 25G 长距离通道参考模型如图 6-1 所示，通道参考差分阻抗为  $100\Omega$ 。测试点位于发射和接收芯片的封装管脚与 PCB 连接处。

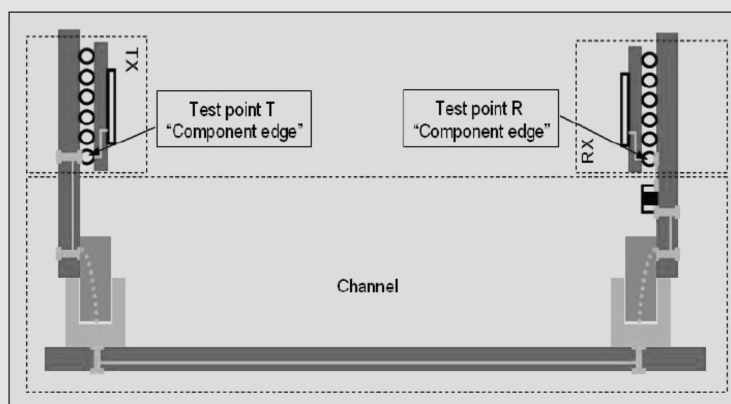


图 6-1 25G 长距离通道参考模型

协议要求直接测量的通道特性包括插入损耗 (IL)、回波损耗 (RL)、近端串扰 (NEXT)、远端串扰 (FEXT) 等，它们的定义分别如表 6-1 所示。

表 6-1 CEI - 25G - LR 通道特性定义 (1)

Symbol	Description
$IL(f)$	Differential insertion loss, - SDD21 magnitude (dB)
$RL_1(f)$	Differential input return loss, - SDD11 magnitude (dB)
$RL_2(f)$	Differential output return loss, - SDD22 magnitude (dB)
$NEXT_m(f)$	Differential near - end crosstalk loss ( $m^{th}$ aggressor), - SDD21 magnitude (dB)
$FEXT_n(f)$	Differential far - end crosstalk loss ( $n^{th}$ aggressor), SDD21 magnitude (dB)

同时, 协议中还规定了根据测量结果计算的通道特性, 包括拟合插入损耗 ( $IL_{fitted}$ )、插入损耗偏差 (ILD)、综合串扰噪声 (ICN)、插入损耗偏差均方根 (ILD<sub>rms</sub>) 等, 它们的定义如表 6-2 所示。

表 6-2 CEI - 25G - LR 通道特性定义 (2)

Symbol	Description
$IL_{fitted}(f)$	Fitted insertion loss (dB)
$ILD(f)$	Insertion loss deviation (dB)
$ICN(f)$	Integrated crosstalk noise (mV, RMS)
ILD <sub>rms</sub>	RMS value of the insertion loss Deviation (dB)



## 6.1 建立通道模型并仿真

(1) 从 ADS 主程序, 打开实例下载资源中提供的工程 CEI0301\_wrk。

(2) 从 Windows 浏览器找到 CEI0301\_wrk 文件夹, 打开文件夹下的 data 子文件夹。该文件夹包含 9 个 .s4p 文件。其中 AGGRESSOR1 ~ 4 为近端串扰的 S 参数测量数据, AGGRESSOR5 ~ 8 为远端串扰的 S 参数测量数据, THRU 为直通通道测量数据, 如图 6-2 所示。

Name	Date modified	Type	Size
AGGRESSOR1	3/2/2011 12:09 AM	S4P File	1,905 KB
AGGRESSOR2	3/2/2011 12:09 AM	S4P File	1,904 KB
AGGRESSOR3	3/2/2011 12:09 AM	S4P File	1,904 KB
AGGRESSOR4	3/2/2011 12:09 AM	S4P File	1,904 KB
AGGRESSOR5	3/2/2011 12:09 AM	S4P File	1,904 KB
AGGRESSOR6	3/2/2011 12:09 AM	S4P File	1,905 KB
AGGRESSOR7	3/2/2011 12:09 AM	S4P File	1,904 KB
AGGRESSOR8	3/2/2011 12:09 AM	S4P File	1,904 KB
THRU	3/2/2011 12:09 AM	S4P File	1,904 KB

图 6-2 仿真用到的 S 参数数据

(3) 创建一个新的原理图 CEI0301test。从原理图元件库中添加元件并连接, 建立如图 6-3 所示的电路。其中, Balun4Port 来自 System - Passive 元件库, DataFileList 和 BatchSim-Controller 来自 Simulation - Batch 元件库。

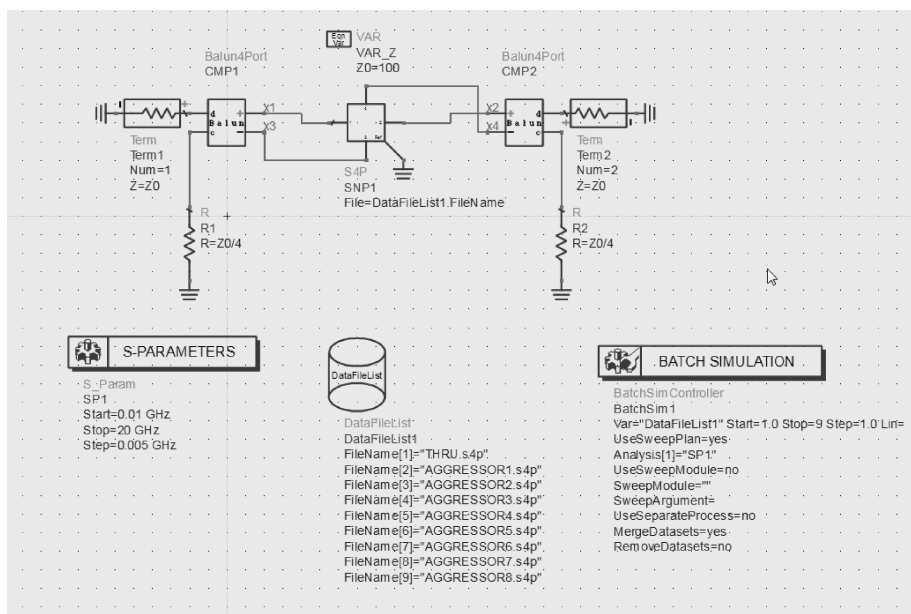


图 6-3 仿真的原理图

(4) 双击 BatchSimController, 在 Sweep 标签栏下选择 Use sweep plan, 单击 Add 添加扫描变量 DataFileList1, 选择扫描类型为 Linear, 扫描范围从 1 到 9, 步进为 1, 如图 6-4 (a) 所示。在 Simulation 标签栏下, 选择 Analysis to run 为 SP1, 如图 6-4 (b) 所示。单击 OK 按钮完成设置, 然后单击快捷键 F7 运行仿真。

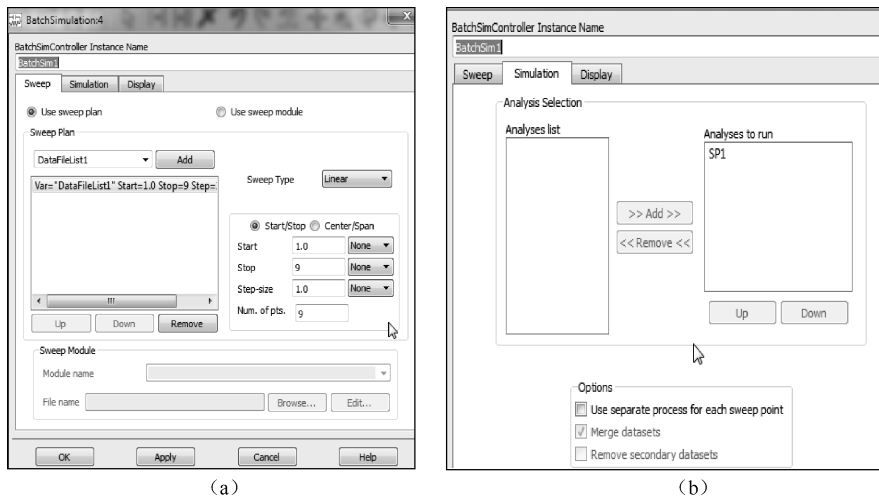


图 6-4 Batch simulation 控制器的设置



## 6.2 通过 AEL 脚本语言自定义测量函数

Application Extension Language (AEL) 是一种类似 C 语言的编程语言。在 ADS 软件中,



AEL 的用途非常广泛。它常用于:

- ☺ 建立用户自定义的元件及组织元件库。
- ☺ 建立参数化的版图设计。
- ☺ 建立用户自定义的功能函数。
- ☺ 控制仿真进程, 实现自动化的仿真, 如批处理仿真、自动优化等。

本实例中, 我们将建立一个自定义的函数 `expandmatrix()`。该函数用于将仿真的 S 参数数组转换为矩阵, 以用于后面的插入损耗曲线拟合。

自定义函数的具体操作步骤如下:

- (1) 新建一个记事本文件, 并命名为 `expandmatrix. ael`。
- (2) 将 `expandmatrix. ael` 保存到 `$HOME\hpeesof\expressions\ael` 文件夹下, 并将以下内容复制到文件中:

```
defun expandmatrix(X)
{
    decl a = sweep_size(X);
    decl b;
    decl newX = X[0];
    decl newY;
    for(b=0;b<a-1;b++)
    {
        newY = {{newX},{X[b+1]}};
        newX = newY;
    }
    return newY;
}
```

`$HOME` 是 ADS 软件默认的工程起始路径。用户可以从 ADS 主窗口选择 `Tools→Configuration Explorer...`, 在列出的系统变量中查找 `HOME` 变量确认对应的 `HOME` 路径。

- (3) 在同一文件夹下, 新建一个记事本文件 `user_defined_fun. ael`, 并在文件中添加以下内容:

```
load("expandmatrix");
```

- (4) 重启 ADS 软件并打开工程。ADS 将在启动过程中自动加载 `HOME` 路径下定义的函数。



## 6.3 插入损耗曲线

- (1) 打开一个新的数据显示窗口, 将默认的数据结果文件设置为 `CEI0301test`。保存数据显示文件为 `CEI0301test. dds`。

- (2) 在数据显示窗口, 选择 `Page→New Page...`, 将显示页面命名为 `insertion loss`。

- (3) 根据协议要求, 通道插入损耗须满足以下要求:

$$IL_{\max} = \begin{cases} 1.083 + 3.35 \sqrt{\frac{f \times 25.8}{f_b}} + 0.96 \frac{f \times 25.8}{f_b}, & f_{\min} \leq f < \frac{f_b}{2} \\ -9.25 + 2.694 \frac{f \times 25.8}{f_b}, & \frac{f_b}{2} \leq f \leq f_b \end{cases}$$

$$IL_{\min} = \begin{cases} 0, & f_{\min} \leq f \leq 1\text{GHz} \\ \frac{1}{3}(f-1), & 1\text{GHz} < f \leq 17.5\text{GHz} \\ 5.5, & 17.5\text{GHz} < f \leq f_b \end{cases}$$

其中,  $f$  为测量频率, 单位 GHz;  $f_{\min}$  为 0.05GHz;  $f_b$  为通道支持的数据速率, 其取值范围为  $19.9\text{Gsym/s} \leq f_b \leq 25.8\text{Gsym/s}$ 。本实例中, 取  $f_b = 20\text{Gsym/s}$ 。

(4) 在 insertion loss 显示页面, 添加如下的等式:

**Eqn** fb=20G

**Eqn** IL=dB(S21)[0,:]

**Eqn** ILmax=if (freq<fb/2) then a else b

**Eqn** a=-(1.083+3.35\*(freq\*25.8/fb)\*\*0.5+0.96\*(freq\*25.8/fb))

**Eqn** b=-(-9.25+2.694\*(freq\*25.8/fb))

**Eqn** ILmin= if (freq<=1GHz) then 0 elseif (freq<=17.5GHz) then -(freq/1e9-1)/3 else -5.5

等式中, dB(S21)[0,: ] 为直通通道 (即 THRU . s4p) 的插入损耗结果。其中, 0 代表 9 组测试数据中指针等于 0 的数据, 即第一组数据。

(5) 在同一页面, 用矩形图显示 IL、IL<sub>min</sub> 和 IL<sub>max</sub> 的数据结果, 如图 6-5 所示。

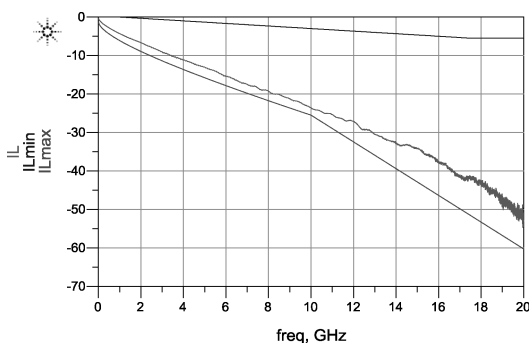


图 6-5 插入损耗曲线



## 6.4 拟合插入损耗曲线

(1) 在数据显示窗口, 选择 Page→New Page..., 将显示页面命名为 Fitted insertion loss。

(2) 根据协议, 测量的插入损耗数据可以拟合一条插入损耗曲线, 拟合的插入损耗曲线须满足以下要求 (见表 6-3)。

表 6-3 拟合插入损耗曲线参数表

Parameter	Units	Value	
		Min	Max
Minimum frequency, $f_{\text{ILmin}}$	GHz	0.05	—
Maximum frequency, $f_{\text{ILmax}}$	GHz	—	$f_b$
Fitted insertion loss at Nyquist	dB	—	25.5
Fitted insertion loss, $a_0$	dB	-1	2.0
Fitted insertion loss, $a_1$	dB	0	20.317
Fitted insertion loss, $a_2$	dB	0	51.6
Fitted insertion loss, $a_4$	dB	0	25.294

拟合插入损耗曲线的表达式如下：

$$\text{IL}_{\text{fitted}}(f) = a_0 + a_1 \sqrt{\frac{f}{f_b}} + a_2 \frac{f}{f_b} + a_4 \left( \frac{f}{f_b} \right)^2 \quad (\text{dB})$$

其中,  $f_b$  为通道支持的数据速率; 多项式系数  $a_0$ ,  $a_1$ ,  $a_2$ ,  $a_4$  的定义为:

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_4 \end{bmatrix} = (F^T F)^{-1} F^T [\text{mag}(\text{IL}_f) \times \text{IL}_f]$$

其中,  $F$  矩阵的定义为:

$$F = \begin{bmatrix} \text{mag}(\text{IL}_{f_1}) & \text{mag}(\text{IL}_{f_1}) \times \sqrt{\frac{f_1}{f_b}} & \text{mag}(\text{IL}_{f_1}) \frac{f_1}{f_b} & \text{mag}(\text{IL}_{f_1}) \times \left( \frac{f_1}{f_b} \right)^2 \\ \text{mag}(\text{IL}_{f_2}) & \text{mag}(\text{IL}_{f_2}) \times \sqrt{\frac{f_2}{f_b}} & \text{mag}(\text{IL}_{f_2}) \frac{f_2}{f_b} & \text{mag}(\text{IL}_{f_2}) \times \left( \frac{f_2}{f_b} \right)^2 \\ \dots & \dots & \dots & \dots \\ \text{mag}(\text{IL}_{f_N}) & \text{mag}(\text{IL}_{f_N}) \times \sqrt{\frac{f_N}{f_b}} & \text{mag}(\text{IL}_{f_N}) \frac{f_N}{f_b} & \text{mag}(\text{IL}_{f_N}) \times \left( \frac{f_N}{f_b} \right)^2 \end{bmatrix}$$

上式中,  $\text{mag}(\text{IL}_{f_N})$  为在测量频率  $f_N$  处的插入损耗幅度。

(3) 在 Fitted insertion loss 显示页面, 创建以下等式:

**Eqn** magIL=mag(S21[0,:])      **Eqn** K=freq[0,:]/fb

**Eqn** M0=magIL

**Eqn** M1=magIL\*(K\*\*0.5)

**Eqn** M2=magIL\*K

**Eqn** M4=magIL\*(K\*\*2)

**Eqn** F1={M0,M1,M2,M4}

**Eqn** F=expandmatrix(F1)      **Eqn** FT=transpose(F)

**Eqn** A=inverse(FT\*F)\*FT\*expandmatrix(M0\*IL)

Eqn  $IL\_fitted=A(1,1)+A(2,1)*K**0.5+A(3,1)*K+A(4,1)*K**2$

Eqn  $min\_A={{-1},{0},{0},{0}}$

Eqn  $max\_A={{2},{20.317},{51.6},{25.294}}$

-A 为计算得到的多项式系数，将 -A 与表 6-4 中标准要求的 min\_A、max\_A 比较，可以看到结果满足标准要求。

表 6-4

-A	min_A	max_A
0.123	-1	2.000
13.351	0	20.317
23.232	0	51.600
9.667	0	25.294

IL\_fitted 为拟合插入损耗曲线，图 6-6 为拟合插入损耗曲线与测量插入损耗曲线 IL 的比较。

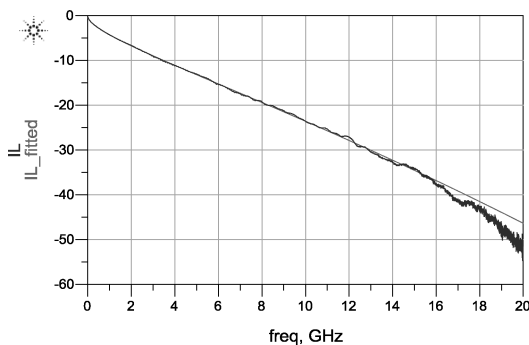


图 6-6 拟合插入损耗曲线与测量插入损耗曲线



## 6.5 插入损耗曲线偏差

- (1) 在数据显示窗口，选择 Page→New Page...，将显示页面命名为 insertion loss deviation。
- (2) 根据协议，插入损耗数据偏差（ILD）等于测量插入损耗曲线与拟合插入损耗曲线之差。

ILD 须满足以下要求：

$$ILD \geq ILD_{\min} = \begin{cases} -1.0 - 12.0(f/f_b), & f_{IL\min} \leq f < f_b/4 \\ -4.0, & f_b/4 \leq f \leq (3/4)f_{IL\max} \end{cases}$$

$$ILD \leq ILD_{\max} = \begin{cases} 1.0 + 12.0(f/f_b), & f_{IL\min} \leq f < f_b/4 \\ 4.0, & f_b/4 \leq f \leq (3/4)f_{IL\max} \end{cases}$$

ILD<sub>rms</sub> 为 ILD 的均方根值，标准中定义的计算公式如下：

$$ILD_{rms} = \sqrt{\frac{\sum W(f) \times ILD(f)^2}{N}}$$

其中， $W(f)$ 的定义如下：

$$W(f) = \text{sinc}^2(f/f_b) \left[ \frac{1}{1 + (f/f_t)^4} \right] \left[ \frac{1}{1 + (f/f_r)^8} \right]$$

式中， $f_t = 0.2365/T_{tr}$ ， $T_{tr}$  为 20% ~ 80% 上升/下降沿时间； $f_r = 3/4f_b$ 。

标准要求  $ILD_{rms}$  须小于 0.3dBrms。

(3) 在 Fitted insertion loss 显示页面，创建以下等式：

**Eqn**ILD=IL-IL\_fitted

**Eqn**ILDmin=if (freq<fb/4) then (-1\*12\*freq/fb) else -4

**Eqn**ILDmax=if (freq<fb/4) then (1+12\*freq/fb) else 4

将计算得到的 ILD、 $ILD_{min}$  和  $ILD_{max}$  在同一张矩形图中比较，如图 6-7 所示。

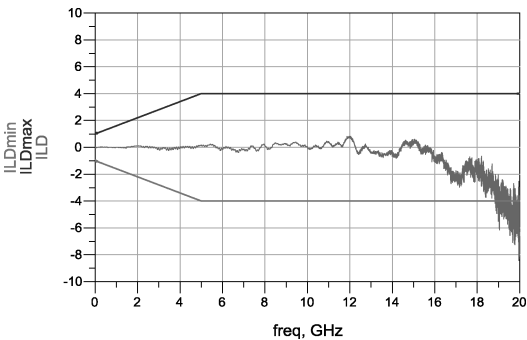


图 6-7 插入损耗数据偏差 (ILD) 测量结果

(4) 在同一显示页面，添加以下等式，并用列表显示计算得到的  $ILD_{rms}$  值。

**Eqn**T\_tr=10ps  
**Eqn**f\_tran=0.2365/T\_tr  
**Eqn**f\_r=fb\*3/4  
**Eqn**W=sinc(freq\*pi/fb)\*\*2\*(1/(1+(freq/f\_tran)\*\*4))\*(1/(1+(freq/f\_r)\*\*8))  
**Eqn**N=sweep\_size(freq[0,:])  
**Eqn**ILD\_rms=(sum(W[0,:]\*ILD\*\*2)/N)\*\*0.5

f_tran	f_r
2.365E10	1.500E10

N
3999

ILD_rms
0.115



## 6.6 回波损耗曲线

(1) 在数据显示窗口，选择 Page→New Page...，将显示页面命名为 insertion loss deviation。

(2) 根据协议，通道回波损耗须满足以下要求：

$$\begin{aligned} RL(f) &\geq 12\text{dB} && \text{当 } f_{min} < f < f_b/4 \\ RL(f) &\geq 12\text{dB} - 15\lg(4f/f_b) && \text{当 } f_b/4 < f < f_b \end{aligned}$$

(3) 在 insertion loss deviation 显示窗口，添加如下等式，并将计算结果与 1、2 端口的回波损耗比较，如图 6-8 所示。

```
Eqn RL_max=if(freq<=fb/4) then -12 else (-12+15*log(4*freq/fb))
```

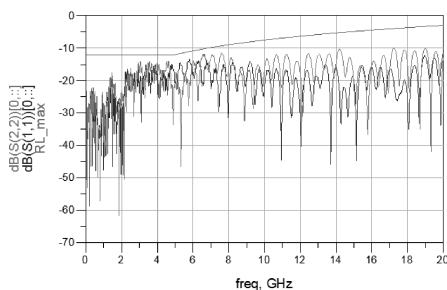


图 6-8 回波损耗测量结果



## 6.7 综合串扰噪声

(1) 在数据显示窗口, 选择 Page→New Page..., 将显示页面命名为 crosstalk。

(2) 协议中, 对通道的综合串扰噪声定义如下:

定义多干扰通道的近端串扰损耗  $\text{MDNEXT}_{\text{loss}}(f)$  如下:

$$\text{MDNEXT}_{\text{loss}}(f) = -10 \times \lg \left( \sum_{i=0}^{\text{all NEXTs}} 10^{-(\text{NLi}(f))/10} \right) (\text{dB})$$

其中,  $\text{NLi}(f)$  为串扰损耗, 单位为 dB;  $f$  为频率, 单位 GHz,  $0.05\text{GHz} < f < f_b$ ;  $i$  表示所有可能的近端串扰通道。

定义多干扰通道的远端串扰损耗  $\text{MDFEXT}_{\text{loss}}(f)$  如下:

$$\text{MDFEXT}_{\text{loss}}(f) = -10 \times \lg \left( \sum_{i=0}^{\text{all FEXTs}} 10^{-(\text{NLi}(f))/10} \right) (\text{dB})$$

其中,  $\text{NLi}(f)$  为串扰损耗, 单位为 dB;  $f$  为频率, 单位 GHz,  $0.05\text{GHz} < f < f_b$ ;  $i$  表示所有可能的远端串扰通道。

定义权函数  $W_{\text{nt}}(f)$  和  $W_{\text{ft}}(f)$  如下:

$$W_{\text{nt}}(f) = (A_{\text{nt}}^2/4f_b) \text{sinc}^2(f/f_b) \left[ \frac{1}{1 + (f/f_{\text{nt}})^4} \right] \left[ \frac{1}{1 + (f/f_r)^2} \right]$$

$$W_{\text{ft}}(f) = (A_{\text{ft}}^2/4f_b) \text{sinc}^2(f/f_b) \left[ \frac{1}{1 + (f/f_{\text{ft}})^4} \right] \left[ \frac{1}{1 + (f/f_r)^2} \right]$$

式中,  $f_{\text{nt}} = 0.2365/T_{\text{ntr}}$ ,  $T_{\text{ntr}}$  为 20% ~ 80% 上升/下降沿时间。  $f_r = 3/4f_b$ 。

近端综合串扰噪声  $\sigma_{\text{nx}}$  定义如下:

$$\sigma_{\text{nx}} = \left( 2\Delta f \sum_n W_{\text{nt}}(f_n) 10^{-\text{MDNEXT}_{\text{loss}}(f_n)/10} \right)^{1/2}$$

远端综合串扰噪声  $\sigma_{\text{fx}}$  定义如下:

$$\sigma_{\text{fx}} = \left( 2\Delta f \sum_n W_{\text{ft}}(f_n) 10^{-\text{MDFEXT}_{\text{loss}}(f_n)/10} \right)^{1/2}$$

全部综合串扰噪声  $\sigma_x$  定位为:

$$\sigma_x = \sqrt{\sigma_{\text{nx}}^2 + \sigma_{\text{fx}}^2}$$

标准要求全部综合串扰噪声满足以下要求：

$$\sigma_x \leq \sigma_{x, \max} \begin{cases} 10 \text{ (mV, RMS)} & 3\text{dB} < \text{IL} \leq 5.3\text{dB} \\ 12.4 - 0.45\text{IL (mV, RMS)} & 5.3\text{dB} < \text{IL} \leq 25.5\text{dB} \end{cases}$$

其中，IL 为通道在  $1/2f_b$  频率处的插入损耗。

(3) 在 cross 显示窗口，添加如下等式及数据列表 (list)。

$$\text{Eqn NL} = \text{dB}(S21)[1:4,:]$$

$$\text{Eqn MDNEXT} = -10 * \log(\text{sum}(10^{**}(\text{permute}(\text{NL})/10)))$$

$$\text{Eqn FL} = \text{dB}(S21)[5:8,:]$$

$$\text{Eqn MDFEXT} = -10 * \log(\text{sum}(10^{**}(\text{permute}(\text{FL})/10)))$$

freq	MDNEXT	MDFEXT
10.00 MHz	44.967	44.500
15.00 MHz	51.899	51.706
20.00 MHz	56.618	57.182
25.00 MHz	54.141	56.511
30.00 MHz	57.936	60.078
35.00 MHz	59.131	59.091
40.00 MHz	59.352	64.064
45.00 MHz	52.540	51.981
50.00 MHz	56.044	52.616
55.00 MHz	54.769	55.223
60.00 MHz	58.803	52.877
65.00 MHz	57.256	55.785
70.00 MHz	54.360	59.760

$$\text{Eqn Ant} = 1200$$

$$\text{Eqn Aft} = 1200$$

$$\text{Eqn T\_nr} = 10\text{ps}$$

$$\text{Eqn f\_nt} = 0.2365/T\_nr$$

$$\text{Eqn T\_fr} = 10\text{ps}$$

$$\text{Eqn f\_ft} = 0.2365/T\_fr$$

$$\text{Eqn Wnt} = (\text{Ant}^{**}2/4/\text{fb}) * \text{sinc}(\text{freq} * \text{pi}/\text{fb})^{**}2 * (1/(1+(\text{freq}/\text{f\_nt})^{**}4)) * (1/(1+(\text{freq}/\text{f\_r})^{**}8))$$

$$\text{Eqn Wft} = (\text{Aft}^{**}2/4/\text{fb}) * \text{sinc}(\text{freq} * \text{pi}/\text{fb})^{**}2 * (1/(1+(\text{freq}/\text{f\_ft})^{**}4)) * (1/(1+(\text{freq}/\text{f\_r})^{**}8))$$

$$\text{Eqn deltaF} = 5\text{MHz}$$

$$\text{Eqn N1} = \text{sum}(\text{Wnt}[0,:]*10^{**}(-\text{MDNEXT}/10))$$

$$\text{Eqn NOISEn} = (2 * \text{deltaF} * \text{N1})^{**}0.5$$

$$\text{Eqn N2} = \text{sum}(\text{Wft}[0,:]*10^{**}(-\text{MDFEXT}/10))$$

$$\text{Eqn NOISEfx} = (2 * \text{deltaF} * \text{N2})^{**}0.5$$

$$\text{Eqn Total\_NOISE} = (\text{NOISEn}^{**}2 + \text{NOISEfx}^{**}2)^{**}0.5$$

NOISEn	NOISEfx	Total_NOISE
2.471	2.181	3.296

$$\text{Eqn IL\_noise} = \text{IL}[\text{find\_index}(\text{freq}[0,:], \text{fb}/2)]$$

IL_noise
-23.574

$$\text{Eqn NOISEmax} = \text{if}(-\text{IL\_noise} > 25.5) \text{ then } 0 \text{ else if } (-\text{IL\_noise} > 5.3) \text{ then } 12.4 - 0.45 * (-\text{IL\_noise}) \text{ else if } (-\text{IL\_noise} > 3) \text{ then } 10 \text{ else } 0$$

$$\text{Eqn noiseresult} = \text{if}(\text{Total\_NOISE} \leq \text{NOISEmax}) \text{ then "PASS" else "FAIL"}$$

NOISEmax	Total_NOISE	noiseresult
1.792	3.296	FAIL

noiseresult 给出了最终仿真噪声结果与标准的比较。

### 【总结】

本实例介绍了使用 ADS 软件对测量的 CEI-25G-LR 通道特性进行一致性分析的方法。

# 实例 7 高速串行链路的通道仿真

## 【目的】

通道仿真 (Channel Simulator) 是分析高速串行链路性能的重要仿真技术。本章将介绍通道仿真器的基本原理及主要设置, 本章还将介绍到 IBIS - AMI 仿真的基本方法。最后, 通过一个 PCIE3.0 串行通道的实例, 说明仿真的流程。

## 【背景知识】

### 1. 通道仿真器的基本原理

**1) 高速串行接口简介** 随着技术的发展, 传统的并行接口技术逐渐被更高速的串行接口技术所取代。SERDES 是英文 SERializer (串行器) /DESerializer (解串器) 的缩写。它在发送端将多路低速并行信号转换成高速串行信号, 经过传输媒通道, 最后在接收端高速串行信号重新转换成低速并行信号。这种通信技术充分利用传输媒体的信道容量, 减少所需的传输信道和器件引脚数目, 从而大大降低了通信成本, 具有功耗低、抗干扰强、速度快的特点。

目前串行总线技术已经成为高速数字信号传输技术的主流, 应用于从芯片端间互联到系统级互联的各个领域。常见的串行总线标准有 PCIeExpress、USB3.0、SAS、HDMI、OIF - CEI - 3.1 光纤通道电接口、802.3ap/ba/bj 以太网等。

**2) 仿真器基本原理** 为了对串行链路进行快速的眼图分析, ADS 提供了专门的通道仿真器。在典型的串行总线场景下, 通道仿真器能够在 1 分钟时间内处理近百万长度的比特序列, 生成精确的眼图, 并进行眼高、眼宽、浴盆曲线、误码率轮廓线等测量。通道仿真充分考虑了实际串行总线中影响信号质量的各种因素, 如码间干扰、抖动、串扰、编码、均衡等。

通道仿真器有两种工作模式可供选择:

**(1) 逐比特模式:** 逐比特模式计算特定 bit 序列的响应。通道仿真器依赖通道线性时不变以达到快速仿真和高吞吐量的要求。

为了理解逐比特模式的工作原理, 我们假设一个输入 NRZ 单比特  $m(t)$ , 它的输出响应为  $p_n(t)$ , 如图 7-1 所示。

假设通道是线性时不变的, 那么对于任意给定 bit 序列

$$x(t) = \sum n r n(t - nT_b)$$

它的输出响应等于其中单比特响应的线性叠加, 即

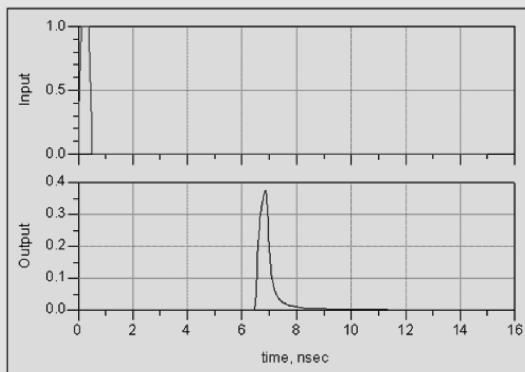


图 7-1 单比特信号的输入与输出波形



$$y(t) = \sum npn(t - nT_b)$$

其中,  $T_b$  为 bit 持续周期, 即 UI。

例如, bit 序列 10101000 的瞬态仿真输出响应与它的单比特脉冲响应叠加结果比较如图 7-2 所示: 可以看到, 传统的瞬态仿真结果与基于单比特脉冲响应叠加的结果是完全吻合的。

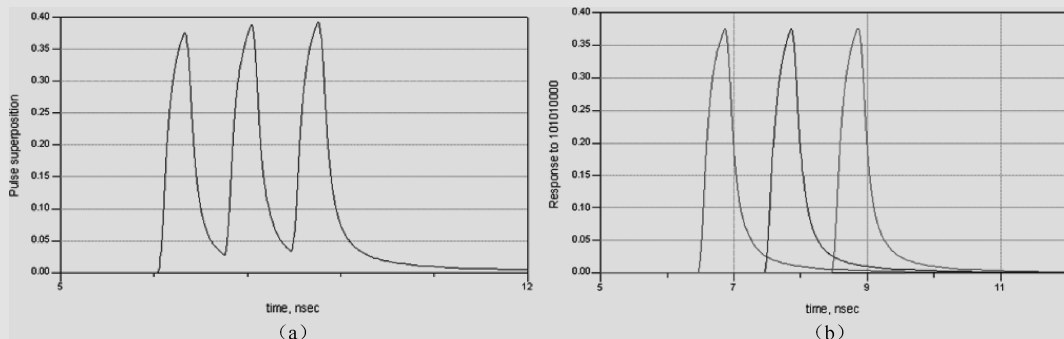


图 7-2 瞬态仿真输出结果 (a) 与单比特响应叠加结果的比较 (b)

在逐比特模式下, 单比特脉冲响应是根据上升沿/下降沿的阶跃响应叠加得到的。

如上所述, 通道逐比特模式仿真包含两个阶段:

- ⑤ 计算阶跃响应: 在这一阶段, ADS 通过瞬态/卷积仿真引擎计算系统阶跃响应, 仿真器自动计算阶跃响应持续时间, 考虑发射器/接收器均衡的影响。
- ⑤ 脉冲响应叠加: 在这一阶段, 仿真器按照输入信号的 bit 序列顺序将单个脉冲响应进行叠加, 将输出结果交给眼图探针进行数据后处理。如果有抖动参数, 则输入信号的上升沿/下降沿将按参数设置进行抖动调制。如果电路中有多个眼图探针和串扰源, 仿真器将重复以上步骤, 直至完成所有测量项。

(2) 统计模式: 在统计模式下, 系统的响应由脉冲响应通过统计计算得到。在这种模式下, 仿真的输出  $y(t) = \sum npn(t - nT_b)$  可视为输入 bit 序列  $b_k$  的函数。在一个 UI 内的某个给定时刻, 波形的概率密度函数  $p(y)$  通过统计计算获得, 如图 7-3 所示。

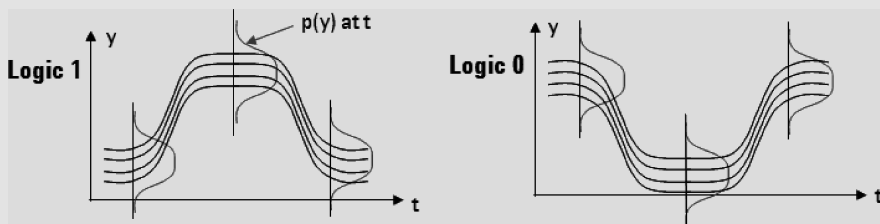


图 7-3 统计模式眼图计算

与逐比特模式一样, 统计模式仿真也包含两个阶段:

- ⑤ 计算阶跃响应: 这一阶段工作原理与逐比特模式完全相同。
- ⑤ 统计计算: 通过统计计算的方法从 ISI 分布产生眼图, 抖动、串扰、均衡、编码等因素在眼图生成中一并考虑。眼图探针元件根据眼图的概率密度分布结果通过后处理进行所有测量项的计算, 包括浴盆曲线、误码率轮廓线等。

(3) 逐比特模式与统计模式的比较：对于不重复的 bit 序列，理论上，当仿真比特数趋于无穷大时，两种仿真模式会给出相同的仿真结果。如果希望得到低误码率下的精确结果，请选择统计模式，如果希望得到对某一特定 bit 序列的仿真结果，请选择逐比特模式。

在逐比特模式中，接收端的 DFE/FFE 均衡器可以工作在自适应模式下。而在统计模式下，DFE/FFE 均衡器不能选择自适应模式，但仍然可以使用固定抽头系数进行仿真。要仿真自适应 DFE/FFE 下的极低误码率结果，可以首先使用逐比特模式仿真足够长比特数，得到稳定的 DFE/FFE 抽头系数并保存，然后切换到统计模式，读入保存的抽头系数进行仿真。

在逐比特模式下，眼图探针可以进行时域波形和时钟信号波形的测量，而统计模式不可以进行以上两项测量。除以上两项测量外，其他测量项两种模式都支持。

下表总结了几种不同的仿真方法的主要区别。

	瞬态仿真	通道仿真 - 逐比特模式	通道仿真 - 统计模式
仿真方法	通过基尔霍夫电流定律计算节点电压电流	脉冲响应的逐比特叠加	基于脉冲响应的统计计算
对电路模型的要求	对通道、发射器、接收器均没有限制	通道必须是线性时不变的，发射器、接收器可以非线性时变	通道、发射器、接收器都必须是线性时不变的
1 分钟内可以仿真的误码率	$\sim 10^{-3}$	$\sim 10^{-6}$	$\sim 10^{-18}$ 或更低

**3) 仿真设置** 要运行通道仿真，需要进行三处设置。

(1) 仿真控制器的设置：包含“Analysis”、“Convolution”两个标签栏。

① 双击仿真控制器打开“Analysis”，如图 7-4 所示。这里设置仿真的模式及控制极低误码率仿真。其中的各项参数意义如下：

- ☺ Bit-by-bit：选择逐比特仿真模式。
- ☺ Statistical：选择统计模式。
- ☺ Number of bits：逐比特模式时的仿真比特数。
- ☺ Enable ultra low BER (<1e-16) simulation：选择这一选项激活极低误码率（仿真误码率 < 1e-16）分析。

☺ Status Levels：用于控制仿真状态窗口显示的信息量。数值越大，显示的信息越详细。默认值为 2，最大值为 5。

② 打开“Convolution”卷积标签栏，如图 7-5 所示。这里的选项用于控制阶跃响应的计算及脉冲响应的叠加（用于逐比特模式）。各项参数意义如下：

- ☺ Tolerance：用于控制阶跃响应计算的速度和精度。
- ☺ Relax：使用宽松的容差控制。仿真速度更快，代价是精度较差。
- ☺ Auto：默认的容差控制，在精度和速度之间有更好的平衡。
- ☺ Strict：严格的容差控制，仿真精度最高但速度较慢。

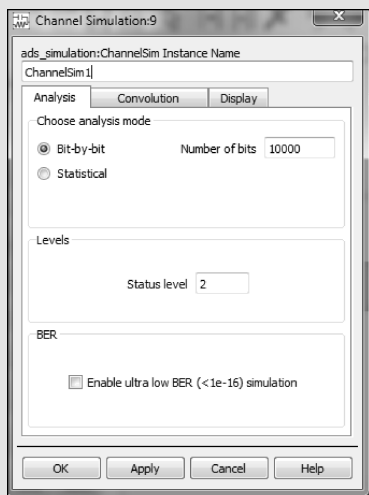


图 7-4 通道仿真器 Analysis 的设置

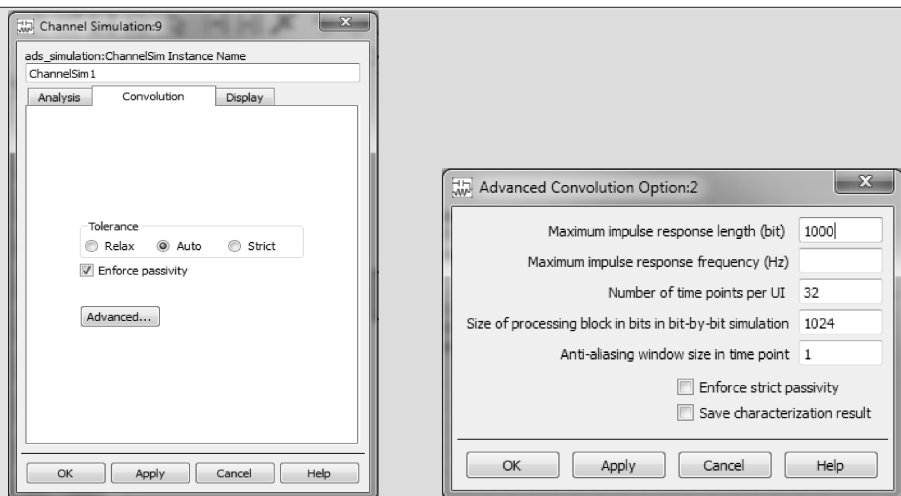


图 7-5 通道仿真器 Convolution 的设置

☺ Enforce Passivity: 在卷积计算中强制阶跃响应无源。

☺ Advanced: 包含一系列优化通道仿真计算的参数。

☺ Maximum impulse response length (bit): 计算冲击响应的最长长度。当计算出的冲击响应无法稳定下来时,可增加这一参数值。默认值为 1000。

☺ Maximum impulse response frequency (Hz): 计算冲击响应时信号频谱的最高频率,默认值由信号的带宽决定。

☺ Number of time points per UI: 冲击响应中一个比特周期内的采样点数。

☺ Size of processing block in bits in bit-by-bit simulation: 逐比特模式仿真中,信号被分成多个等长的数据段分批次进行处理。这一参数设置分割的数据段长度。

☺ Anti-aliasing window size in time points: 带通信号反关联窗的大小。用于对带通信号进行滤波。

(2) 发射器 (TX)、接收器 (RX) 和串扰 (Xtalk) 的设置。

① ADS 的通用通道发射器包含 Tx\_SingleEnded 和 Tx\_Diff 两个元件。两者的参数设置是一样的,区别是:一个是单端发射器,一个是差分发射器。发射器的参数设置包含 5 个标签栏,如图 7-6 所示。

☺ PRBS: 设置发射端激励源。

Bit rate: 信号源比特率,这一参数也用于设置仿真的比特率。

Vhigh: 逻辑 1 电平电压值。

Vlow: 逻辑 0 电平电压值。

Rise/Fall time: 信号源上升沿/下降沿时间。

Mode: 伪随机序列模式。它有四种模式可供选择:

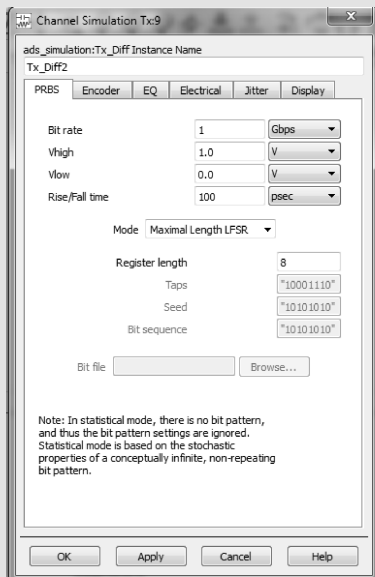


图 7-6 通道发射器 Tx\_Diff 的设置

Maximal Length LFSR

User Defined LFSR

User Defined Sequence

Bit File

Register length: 当“Mode”选择为 Maximal Length LFSR 时, 设置移位寄存器长度。

Taps 和 Seeds: 当“Mode”选择为 User Defined LFSR 时, 设置移位寄存器的抽头系数和种子值。

Bit sequence: 当“Mode”选择为 User Defined Sequence 时, 指定用户自定义的比特序列。

Bit file: 当“Mode”选择为 Bit File 时, 指定用户自定义的外部比特文件。比特文件为文本文件, 内容为任意的 1 和 0 的比特序列。

☺ Encoder: 设置加密算法。

No encoder: 默认选项, 不进行加密。

8B10B: 采用 8B/10B 编码加密。

64B66B: 采用 64B/66B 编码加密。

128B130B: 采用 128B/130B 编码加密。

☺ EQ: 设置发射端加重滤波器。

Choose equalization method: 选择均衡模式 Specify FIR taps、Specify de-emphasis 或者 None。

De-emphasis (dB): 当均衡模式为 Specify de-emphasis 时设置去加重系数, 单位 dB。

Pre Cursor: 当均衡模式为 Specify de-emphasis 时, 设置 PreCursor[n] 的抽头系数。

Post Cursor: 当均衡模式为 Specify de-emphasis 时, 设置 PostCursor[n] 的抽头系数。

Tap Interval (UI): 设置抽头延时, 单位为 UI。

☺ Electrical: 设置源阻抗。

Load: 设置源阻抗值, 默认为  $50\Omega$ 。

Exclude load: 选中这一选项时, 信号源内阻为 0。

☺ Jitter: 设置抖动参数。

DCD (UI): 设置占空比失真, 单位为 UI。

Clock DCD (UI): 设置时钟信号占空比失真, 单位为 UI。

PJ amplitude: 周期性抖动幅度, 单位为 s。

PJ frequency: 周期性抖动频率, 单位为 Hz。

RJ (UI): 随机抖动均方根值, 单位为 UI。

② ADS 的通用通道接收器也包含 Rx\_SingleEnded 和 Rx\_Diff 两个元件, 两者的参数设置是一样的。接收器的参数设置包含 3 个标签栏, 如图 7-7 所示。

☺ EQ: 设置连续时间均衡 (CTLE)、前馈均衡 (FFE) 和判决反馈均衡 (DFE)。

Enable Continuous – time linear equalizer: 激活 CTLE 均衡。单击 Edit 可进入到 CTLE 设置窗口, 如图 7-8 所示。

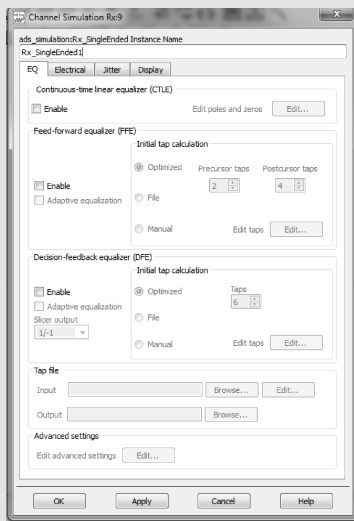


图 7-7 通道接收器 Rx\_Diff 的设置

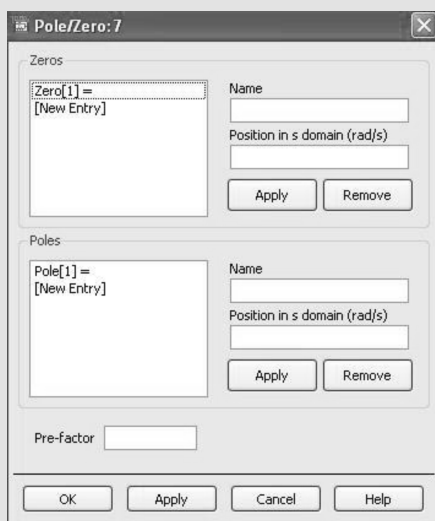


图 7-8 CTLE 均衡的设置

Zero[n]: 在 S 域指定第 n 个零点的位置, 单位为 rad/s。

Pole[n]: 在 S 域指定第 n 个极点的位置, 单位为 rad/s。

Pre – factor: 增益因子。

Enable Feed – forward equalizer: 激活前馈均衡。

Adaptive equalization: 打开自适应 FFE。

Optimized: 自动优化抽头系数, 使眼高最大。

Precursor taps: 优化模式下 pre – cursor 抽头系数数量。

Postcursor taps: 优化模式下 post – cursor 抽头系数数量。

File: 抽头系数从外部文件读入。

Manual: 手动制定抽头系数。

Enable Decision – feedback equalizer: 激活判决反馈均衡。

Adaptive equalization: 打开自适应 DFE。

Optimized: 自动优化抽头系数, 使眼高最大。

Slicer output: 设置 DFE 限幅器的输出为 1/ –1 或 1/0。

Taps: 优化模式下抽头系数数量。

File: 抽头系数从外部文件读入。

Manual: 手动制定抽头系数。

⊙ Electrical: 设置接收器负载阻抗。

Load: 设置负载阻抗值, 默认为 50Ω。

Exclude load: 选中这一选项时, 负载内阻为无穷大。

⊙ Jitter: 设置接收器注入随机抖动。

RJ(UI): 随机抖动均方根值, 单位为 UI。

Amplitude noise(V): 接收端随机噪声均方根值, 单位为 V。

③ ADS 的通用通道串扰元件也包含 Xtlk2\_SingleEnded 和 Xtlk2\_Diff 两个元件, 两者的参数设置是一样的。串扰的参数设置如图 7-9 所示, 包含 5 个标签栏, 其设定与发射器基本相同。唯一的区别在 PRBS 部分, 与发射器相比多出以下几个选项:

- ☺ Phase relative to TX: 与发射器的相对相位关系。
- ☺ Random: 串扰源相位与发射源具有随机(异步)关系。
- ☺ Fixed: 串扰源相位与发射源具有固定(同步)关系。
- ☺ Phase to TX: 串扰源与发射源的固定相位差。

(3) 眼图探针(Eye\_Probe)的设置: 眼图探针用于收集仿真数据并对数据结果进行处理。ADS 中的眼图探针包括单端眼图探针(Eye\_Probe)和差分眼图探针(EyeDiff\_Probe)两种, 两者的内部参数设置是一样的。

眼图探针设置包含 Parameters 和 Measurements 两个标签栏, 如图 7-10 所示。

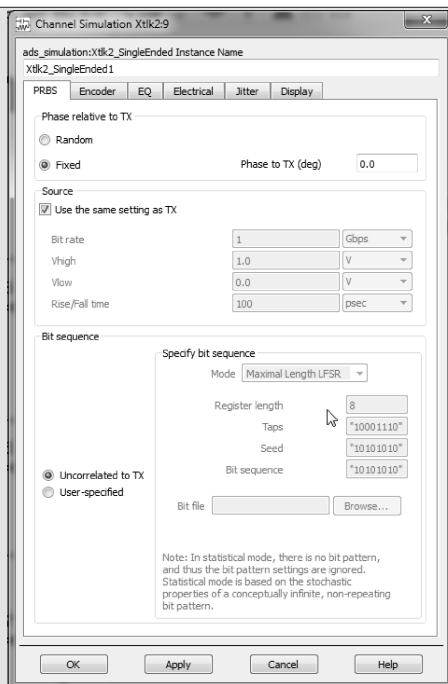


图 7-9 通道串扰元件的设置

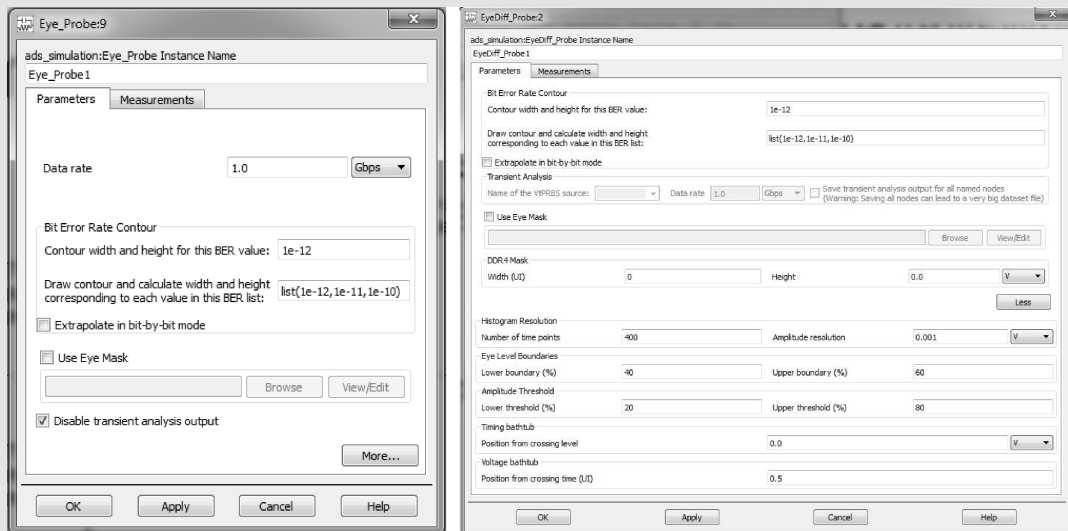


图 7-10 眼图探针的设置

① Parameters 和 Measurements 标签栏下的参数意义如下:

- ☺ Data Rate: 信号数据率。
- ☺ Contour width and height for this value: WidthAtBER 和 HeightAtBER 测量所对应的误码率轮廓线值, 默认为  $1e-12$ 。



- ☺ Draw contour and calculate width and height corresponding to each value in this list: 需要计算误码率轮廓线所对应的误码率列表, 默认值为 list (1e-12, 1e-11, 1e-10)。
- ☺ Extrapolate in bit-by-bit mode: 在逐比特模式下对误码率进行外插。
- ☺ Use eye mask: 使用眼图模板并指定眼图模板路径。
- ☺ Disable transient analysis output: 在 transient 仿真中不保存节点时域波形。
- ☺ Number of time points: 眼图显示时, 横轴 (2 个 UI) 内的时间点数。
- ☺ Amplitude resolution: 眼图显示时, 纵轴的幅度分辨率。
- ☺ Lower boundary (%): 电平值计算时的下边界, 默认为 40%。
- ☺ Upper boundary (%): 电平值计算时的上边界, 默认为 60%。
- ☺ Lower threshold (%): 上升沿/下降沿计算时的低幅度门限, 默认为 20%。
- ☺ Upper threshold (%): 上升沿/下降沿计算时的高幅度门限, 默认为 80%。
- ☺ Position from crossing level: 时间浴盆曲线测量时, 测量幅度与眼图交叉电平的偏移值。
- ☺ Position from crossing time: 幅度浴盆曲线测量时, 测量时间与眼图交叉时间之间偏移值。

② Measurements 标签栏设置所有可能的眼图测量项。按照测量类型的不同, 可以将所有的测量项分为 4 类, 即垂直测量 (信号幅度、眼高、高电平、低电平、信噪比等)、水平测量 (上升沿/下降沿时间、抖动、眼宽等)、误码率测量 (轮廓线、浴盆曲线等) 和其他测量 (眼图、波形、眼图模板、时钟信号等)。

☺ 垂直测量。垂直测量的测量项包括:

Level1: 垂直测量中, 眼图上半部分边界内幅度的平均值。

Level0: 垂直测量中, 眼图上半部分边界内幅度的平均值。

Amplitude: level1 与 level0 之间的差值。

Height: 眼高值, 定义为 level1 与 level0 之间幅度分布的 3sigma 点之间的距离, 如图 7-11 所示。Height 的计算公式如下:

$$\text{Height} = (\text{Level1} - 3 * \text{sigmalevel1}) - (\text{Level0} + 3 * \text{sigmalevel0})$$

HeightDB: 以 dB 表示的眼高, 等于  $10 * \log_{10}(\text{Height})$ 。

LevelMean: Level1 与 Level0 的平均值。

SNR: 信噪比, 计算公式为:  $\text{SNR} = \text{Amplitude} / (\text{sigmalevel1} + \text{sigmalevel0})$ 。

☺ 水平测量。水平测量的测量项包括:

JitterPP: JitterPP 定义为眼图中最大交叉时间与最小交叉时间之差。在统计模式中, 眼图是按照运行 1e6 比特等效做出的。需要注意的是, JitterPP 只有在确定性抖动 (如码间干扰 ISI 或正弦抖动) 占主要地位时才有意义。请不要将这一测量与给定误码率下根据双狄拉克模型计算出的抖动峰-峰值测量混淆。

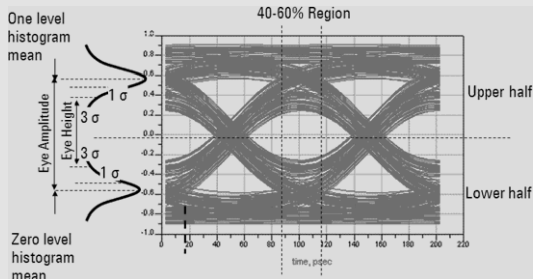


图 7-11 眼图探针的垂直测量示意图

JitterRMS: JitterRMS 定义为交叉时间概率的标准差。

RiseTime 和 FallTime: 测量高电平与低电平幅度门限之间的时间差。高低电平的门限默认值分别由眼图探针的 Amplitude Threshold 控制, 默认值为 Amplitude 测量项的 80%, 20%。 $\text{Amplitude} = \text{Level1} - \text{Level0}$ 。

Width: 眼宽。眼宽的计算公式为:  $\text{Width} = (t2 - 3 * \text{sigmat2}) - (t1 + 3 * \text{sigmat1})$ 。其中  $t2$  和  $t1$  为交叉时间概率的平均值, 如图 7-12 所示。

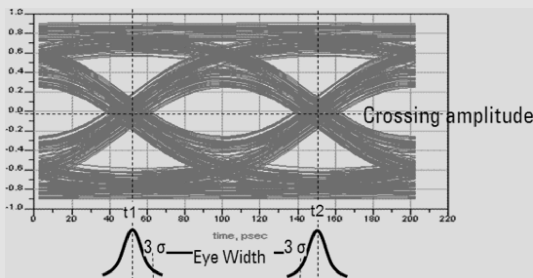


图 7-12 眼图探针的水平测量示意图

☺ 误码率相关的测量项包括:

Contour: 误码率轮廓线。这一测量项测量给定误码率的概率轮廓线, 输出结果 BERContour 保存至仿真结果文件。在眼图探针元件中可以指定任意误码率的轮廓线测量。误码率轮廓线的测量都是在一个比特周期 (UI) 之内进行的。如果计算的误码率低于系统能够计算的最低误码率, 仿真器会给出警告。

Bathtub: 这一测量计算时间与幅度浴盆曲线, 在仿真结果文件中分别保存为 TimingBathtub 和 VoltageBathtub。时间浴盆曲线的参考测量电平默认设置为眼图的交叉电平 (即图 7-12 所示的 crossing amplitude)。同样, 电压浴盆曲线的参考测量时间默认设置为距离眼图高低电平交叉时间 (即图 7-12 所示的  $t1$ 、 $t2$ ) 半个比特周期的时刻。

WidthAtBER 和 HeightAtBER: 计算给定误码率下根据误码率轮廓线得到的最大眼宽与眼高。

☺ 其他测量项包括:

Density: 显示横轴为 2 个 UI 的眼图结果。眼图的颜色表示电平出现的概率密度, 蓝色表示密度较小, 红色表明密度较大。

Waveform: 保存眼图探针所在节点的时域波形。

CheckMaskViolation: 将眼图与指定的眼图模板进行比较, 将比较结果返回给测量值 MaskViolated。当眼图与眼图模板冲突时,  $\text{MaskViolated} = 1$ , 否则  $\text{MaskViolated} = 0$ 。

ClockSignal: 产生时钟信号。

## 2. IBIS - AMI 仿真

1) IBIS - AMI 简介 SERDES 总线的均衡及时钟恢复电路是在发射/接收芯片的内部实现的。芯片的 I/O 模型通常通过 IBIS 模型提供给用户, 而传统的 IBIS 模型无法对复



杂的均衡算法电路进行描述。同时,芯片厂商出于知识产权保护的目的,通常也不会提供详细的 SPICE 电路模型给用户使用。即使用户拿到芯片的 SPICE 模型,要直接通过 SPICE 仿真分析并优化低误码率下的系统性能也因为需要花费太长的时间而变得完全不现实。

为了应对以上新的仿真需求,IBIS 开放论坛在 2008 年 8 月的 IBIS5.0 规范中提出了新的仿真流程解决方案。IBIS 开放论坛为这一解决方案设定了 6 大目标:

- ☺ 知识产权保护:芯片厂家只提供模型算法部分的加密动态链接库文件,以避免逆向设计。
- ☺ 可移植性:同一 IC 模型可以在不同 EDA 厂家的仿真器上使用。
- ☺ 兼容性:来自不同 IC 厂家的模型可以在同一电路中仿真。
- ☺ 高性能:可以在秒级时间内仿真极低误码率轮廓线。
- ☺ 灵活性:新的仿真流程须支持两种仿真模式:统计仿真与时域仿真。统计仿真要求算法模型满足线性时不变要求,而时域仿真可以分析非线性、时变的算法模型。
- ☺ 可优化:模型提供外部可控的参数(如抽头系数等)。用户可以通过电路仿真器对参数进行扫描、优化,改善系统性能。

IBIS 开放论坛将这一新的方案接口称为“AMI”。AMI 是 Algorithmic Modeling Interface 的缩写。AMI 将 SERDES 设备模型分为电气和算法两个部分。电气部分包括发射端与接收端芯片 I/O 模拟电路部分及无源通道,AMI 要求它是线性时不变的,其中芯片的 I/O 模拟电路部分通过传统的 IBIS 模型实现。算法部分包括发射端和接收端均衡算法及时钟恢复电路,对这一部分不要求必须是线性时不变。算法部分的具体实现通过动态链接库文件封装为一个黑盒子,只通过 .ami 的参数文件与 IBIS 文件实现对接。

一个 AMI 模型通常包含以下几个文件:

- ☺ .ibs 文件:传统的 .ibs 文件,包含模拟电路接口以及 [Algorithmic Model] 关键词部分。[Algorithmic Model] 关键词部分定义了 AMI 模型的另外几个文件(.ami、.dll)路径以及编译平台。
- ☺ .ami 文件:指定 AMI 模型参数。
- ☺ .dll 或 .so 文件:定义均衡及时钟恢复算法。

**2) IBIS-AMI 仿真设置** 在 ADS 软件中,AMI 仿真流程要求的两种仿真模式:统计仿真与时域仿真,都可以通道仿真器实现。其中,时域仿真模式在 ADS 通道仿真器中称为逐比特 (“bit-by-bit”) 仿真模式。

如前一节所述,两种仿真模式对 AMI 模型算法部分的要求有所不同:统计模式要求算法模型满足线性时不变要求,而逐比特模式没有限定。因此,并非所有的 AMI 模型都可以使用统计模式进行仿真。

进行 AMI 仿真需要专门的 AMI 元件。ADS 中提供了 6 种 AMI 元件:

- ☺ Tx\_AMI:发射端。
- ☺ Rx\_AMI:接收端。
- ☺ XtlkTx\_AMI:串扰发射端。
- ☺ XtlkRx\_AMI:串扰接收端。
- ☺ Retimer 和 Redriver:中继器。

图 7-13 所示的是典型的 IBIS - AMI 仿真原理图。除了通道的发射端、接收端和串扰端使用了 AMI 元件之外，它的设置与常规的通道仿真基本相同。

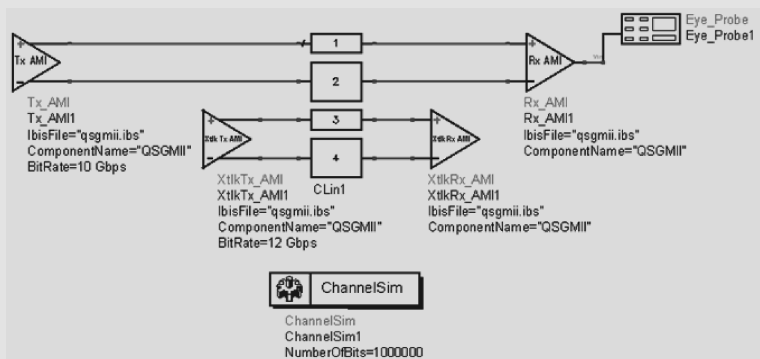


图 7-13 IBIS AMI 仿真原理图

双击 Tx\_AMI 元件，如图 7-14 所示，里面的各项参数意义如下。

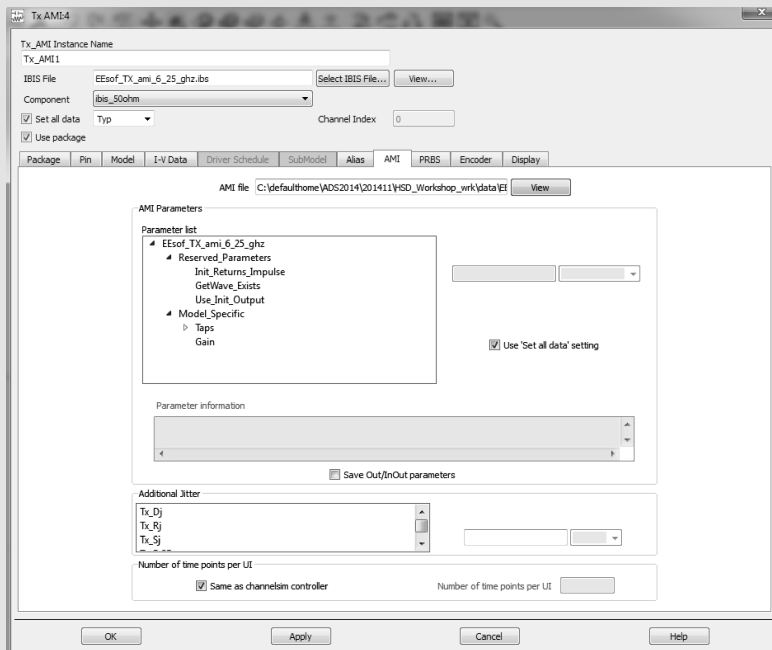


图 7-14 Tx\_AMI 元件的参数设置

- ☺ IBIS File: 指定 IBIS 文件名。
- ☺ Component: 下拉选择要仿真的芯片名称。
- ☺ Set all data: 下拉选择 IBIS 模型的 corner 值 (Typical, Min, Max, Fast, Slow)。
- ☺ Use Package: 是否使用 RLC 封装参数。
- ☺ Package, Pin, Model, I\_V Data, Driver Schedule, SubModel: 用于指定仿真的 IBIS 管脚、模型等。
- ☺ AMI: 设置 AMI 模型参数:
  - Reserved Parameters: IBIS 规范中定义的保留参数。

Model Specific: 规范中没有的, 自定义的模型参数。

Additional Jitter: 除了模型中定义的抖动参数之外的附加抖动。

☺ PRBS, Encoder: 设置发射源的码型及编码方式, 请参考上文关于通道发射器的参数介绍。

☺ Display: 控制前述参数是否在桌面上显示。

常见的 IBIS 规范中 AMI 保留参数有:

☺ AMI\_Version: AMI 模型版本。

☺ Init\_Returns\_Impulse: 若为 True, 算法模型有等效的线性时不变脉冲响应, 此时 AMI 模型可用于统计模式; 若为 False, 则算法模型没有等效的线性时不变脉冲响应, 不能用于统计模式。Init\_Returns\_Impulse = True 常见于发射端模型, 或简单的固定系数均衡、没有时钟恢复的接收端模型。

☺ GetWave\_Exists: 若为 True, 算法模型一般为非线性时变模型, 需要通过 AMI\_GetWave 函数生成输出信号。GetWave\_Exists = True 常见于采用自适应均衡、带有时钟信号恢复功能的接收端模型。

通常, Init\_Returns\_Impulse 和 GetWave\_Exists 取值相反。若 Init\_Returns\_Impulse = False, 则 GetWave\_Exists = True; 若 Init\_Returns\_Impulse = True, GetWave\_Exists = False。Init\_Returns\_Impulse 和 GetWave\_Exists 均为 True 的情况极少。

☺ Use\_Init\_Output: 若为 True, 仿真器创建提交给 AMI\_GetWave 的输入信号时使用 AMI\_Init 函数输出的脉冲响应; 若为 False, 仿真器创建提交给 AMI\_GetWave 的输入信号时使用通道的原始脉冲响应。

☺ Max\_Init\_Aggressors: 仿真允许的最大串扰通道数。

☺ Ignore\_Bits: 用于控制时域仿真(逐比特仿真)时忽略的比特数。当 AMI 模型为时变模型时, 仿真器需要通过 AMI\_Getwave 函数完成均衡的初始化, 这一初始化过程中仿真的输出将被忽略掉。

自定义的参数通常用于控制均衡算法。可选择自定义参数的输入方式为 User specified, 如图 7-15 所示手动输入数值或通过外部定义的变量设置数值, 还可进行参数扫描与优化。

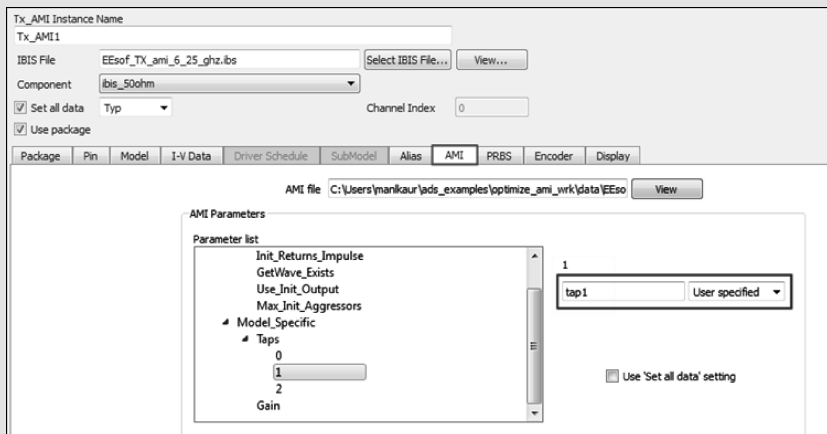


图 7-15 通过变量设置用户自定义 AMI 参数

其他 AMI 元件的参数设置与 Tx\_AMI 元件类似, 这里不再赘述。



## 7.1 PCIE 通道模型

- (1) 从 ADS 主程序，打开实例下载资源中提供的 PCIE\_8G\_wrk 工程。
- (2) 打开设计单元 a\_pcie\_channel\_sim 下的原理图。该原理图如图 7-16 所示。

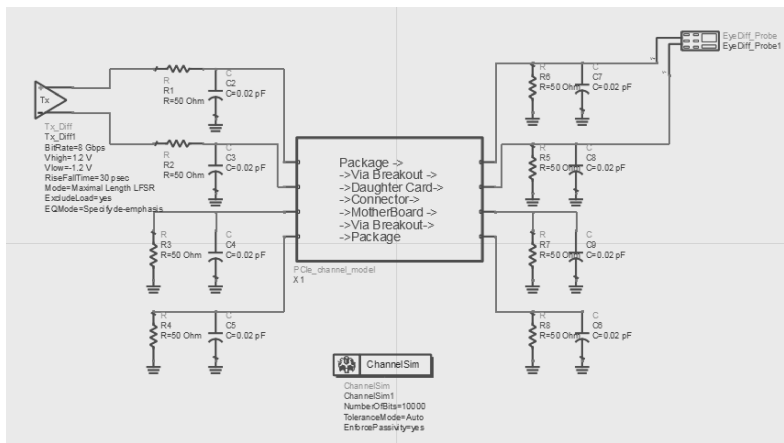

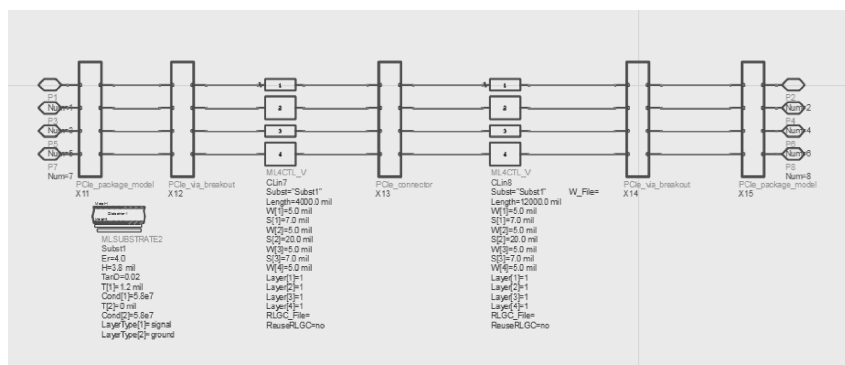
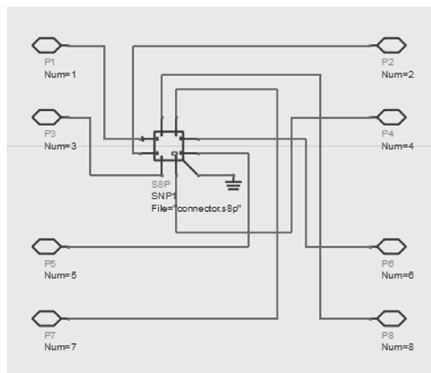


图 7-16 PCIE 通道仿真原理图

- (3) 选择原理图中央的 PCIe\_channel\_model 元件，单击 Push Into Hierachy 命令 ，进入元件的底层电路，如图 7-17 所示。完整的 PCIe 通道模型由芯片封装、过孔 breakout、子板



(a)



(b)

图 7-17 PCie\_channel\_model (a) 和 PCie\_connector 的内部电路 (b)

走线、连接器、主板走线等部分组成。图中，子板与主板的走线部分通过 multi-layer 传输线进行建模；芯片封装与过孔 breakout 的 S 参数模型通过电磁场仿真提取获得；连接器部分的 S 参数模型则由测量获得。选择 PCIe\_connector 元件，单击 Push Into Hierachy 命令 可进一步查看连接器的 S 参数模型。



## 7.2 基本的通道仿真

- (1) 单击 Pop Out 命令 退回到原理图 a\_pcie\_channel\_sim。
- (2) 如图 7-18 所示，双击发射器元件 Tx\_Diff1，在 PRBS 标签栏下，设置发射端的比特速率为 8Gbps，Vhigh = 1.2V，Vlow = -1.2V，上升/下降沿时间为 30ps，移位寄存器长度为 23。在 EQ 标签栏下，设置均衡方法为 Specify de\_emphasis，去加重系数为 3.5dB。

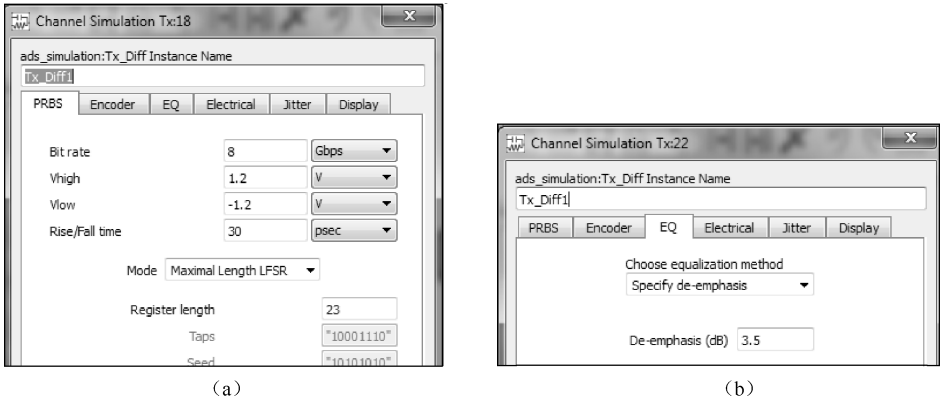


图 7-18 发射器元件 Tx\_Diff1 的设置

- (3) 设置通道仿真器的工作模式为逐比特模式 (bit-by-bit)，仿真的比特数为 10000。
- (4) 保持眼图探针元件采用默认设置。注意，眼图探针中默认设置的测量数据速率为 1Gbps，这不会影响仿真结果的准确性。ADS 会在仿真过程中自动将眼图探针的测量速率设置为与发射器相同的速率。
- (5) 运行仿真。仿真完毕后，打开数据显示窗口 a\_pcie\_channel\_sim.dds，查看对应的眼图及眼高、眼宽 (见图 7-19)。

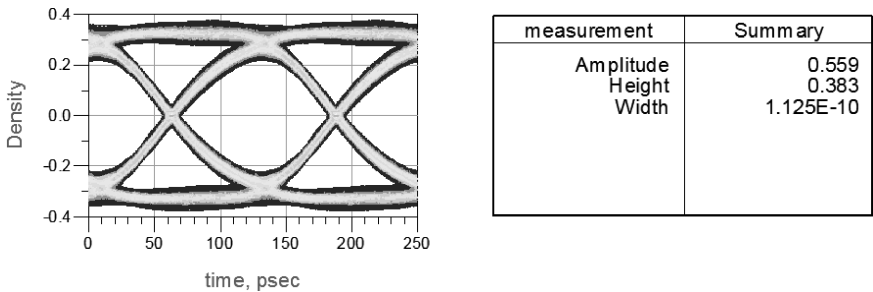


图 7-19 仿真的眼图及测量项结果



## 7.3 串扰 (Xtalk) 对通道性能的影响

(1) 打开 b\_pcie\_channel\_xtlk\_sim 设计单元下的原理图 (见图 7-20)。这一原理图与之前原理图的唯一区别是, 在右下角位置, 原有的端接负载处添加了 Xtlk2\_Diff 的串扰信号源。

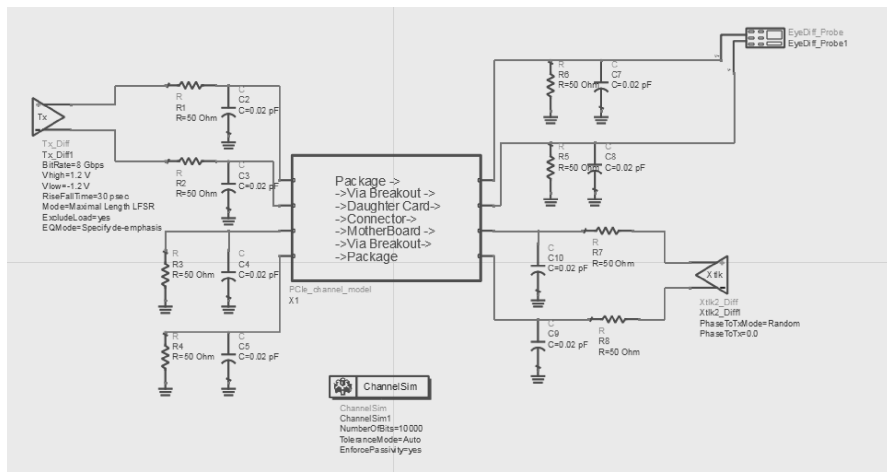


图 7-20 串扰仿真原理图

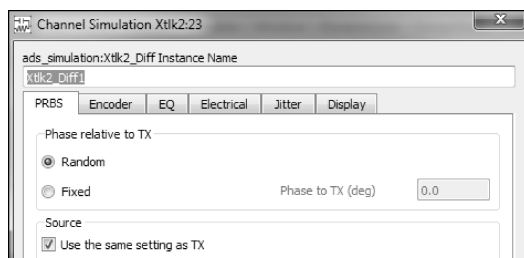


图 7-21 串扰源的设置

(2) 双击 Xtlk2\_Diff 串扰信号源, 它的设定与 TX 发射器基本相同 (见图 7-21)。用户可以设定串扰的数据速率、相位及比特序列与 TX 的关系。此处采用默认设置。如果要研究串扰源相位与 TX 发射器相位关系对信号质量的影响, 可以设定串扰源与 TX 的相对相位关系为 Fixed, 并扫描相对相位值。

(3) 运行仿真。仿真完成后查看结果。与没有串扰信号的仿真结果比较, 可以看到眼图的眼高和眼宽均有所恶化 (见图 7-22)。

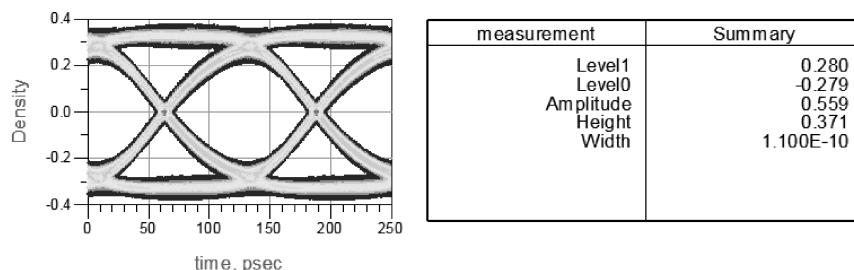


图 7-22 串扰仿真结果



## 7.4 CTLE 均衡

(1) 打开原理图 c\_pcie\_no\_eq\_sim。这一设计中的 TX 与 RX 元件都没有设定均衡。运

行仿真，得到的眼图及眼高、眼宽，如图 7-23 所示。

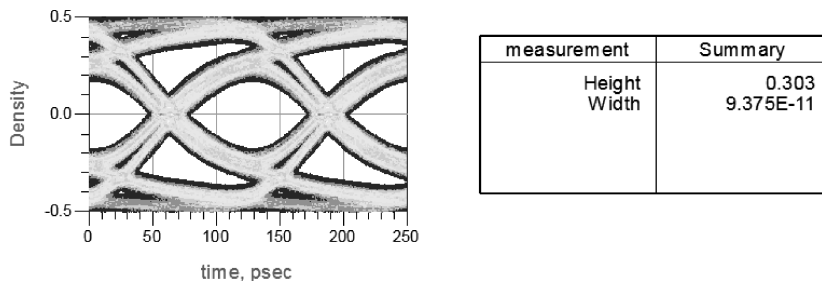


图 7-23 没有均衡时的眼图结果

(2) 打开数据显示文件 d\_Equalizer\_Transfer\_Function. dds，翻至第 2 页，Tune\_CTLE\_for\_Channel\_Loss。这一页面演示了如何进行 CTLE 滤波器零极点的选择。假设要设计的 CTLE 滤波器的传递函数为：

$$H(s) = \text{Pre\_factor} * \frac{(s - z_1)}{(s - p_1)(s - p_2)}$$

式中， $z_1$  为传输零点； $p_1$ 、 $p_2$  为传输极点；Pre - factor 为增益因子，当  $s=0$  时，Pre - factor =  $H(0) * p_1 * p_2 / (-z_1)$ 。 $H(0)$  为直流增益，一般等于 1，此时 Pre - factor =  $p_1 * p_2 / (-z_1)$ 。

在数据显示窗口中，通过拖动 marker 点 m5 的位置，可以选择传输零点。同样，拖动 m6 和 m7 选择两个传输极点，m8 选择增益常数的倒数。

等式 tune\_Transfer\_function 计算了以 dB 为单位表示的传递函数  $H(s)$  的倒数。由于 CTLE 滤波器是作为传输通道插损的补偿，所以它的传递函数的倒数应当近似等于传输通道的插损。

在图 7-24 左下角的直角坐标数据图中， $\text{dB}(\$channel\_sparameters\_loss..S(2,1))$  为仿真得到的通道的插损曲线。拖动 m5、m6、m7，使 tune\_Transfer\_function 的曲线与插损曲线尽量逼近，从而确定最优的零极点。

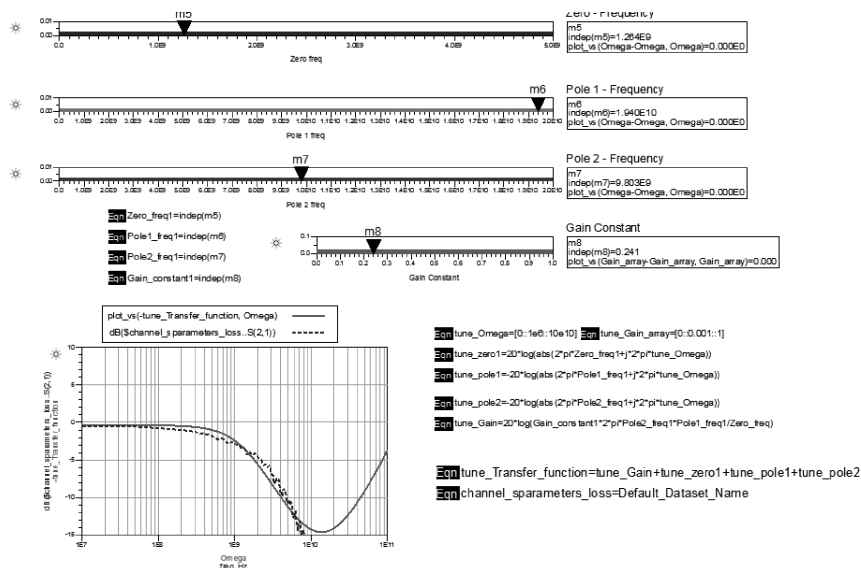


图 7-24 CTLE 均衡的传递函数计算

本实例中，确定的零极点位置为： $z1 = 1.26G$ ， $p1 = 19.4G$ ， $p2 = 9.803G$ 。  
这一结果将用于下一步的原理图设计中。

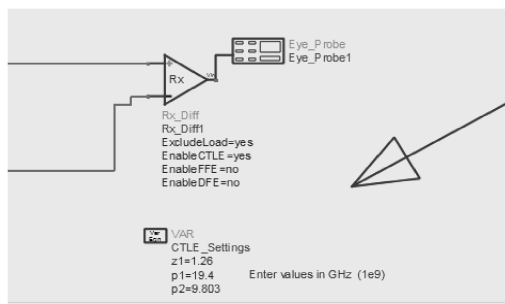


图 7-25 原理图 d\_pcie\_ctle2\_sim

这里  $\pi$  是 ADS 的内置变量， $\pi = 3.1415926$ 。 $z1$ 、 $p1$ 、 $p2$  是原理图中定义的变量，变量值如图 7-26 所示。设置 Pre-factor 的数值等于  $(2 * \pi * p1 * 1e9) * (2 * \pi * p2 * 1e9) / (2 * \pi * z1 * 1e9)$ 。单击 OK 按钮完成设置（见图 7-26）。

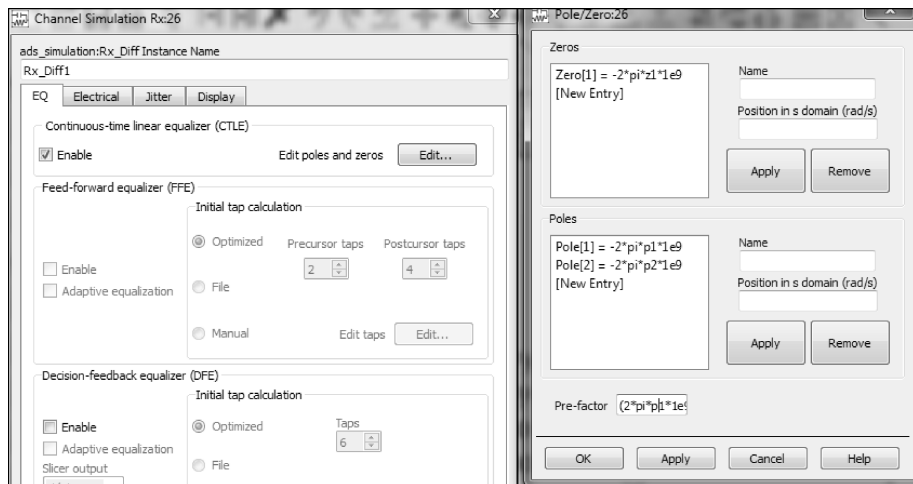
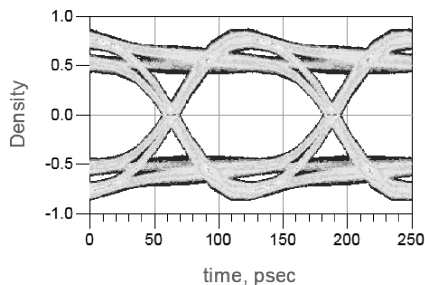


图 7-26 CTLE 均衡的设置

运行仿真，经过 CTLE 均衡后的眼图、眼高、眼宽如图 7-27 所示。由于 CTLE 对高频插损有很大的补偿，所以可以看到均衡后的眼高与均衡前相比明显增加。



measurement	Summary
Level1	0.667
Level0	-0.667
Height	0.850
Width	1.119E-10

图 7-27 经过 CTLE 均衡后的眼图





## 7.5 FFE 均衡

(1) 如图 7-28 所示，打开原理图 e\_pcie\_channel\_ffe\_sim，这一设计与前一设计相比除了 Rx\_Diff 部分之外没有任何不同。如图 7-29 所示，打开 RX\_Diff 设置窗口，在 EQ 标签栏下，Feed - forward equalizer 部分选择 Enable，设置初始的抽头系数计算方式为 optimized 并设置 precursor tap = 1，postcursortap = 2。在 Tap file 部分，设置输出抽头系数文件（Output）到“taps.txt”。

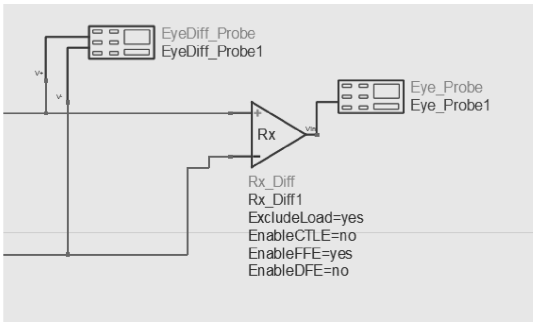


图 7-28 原理图 e\_pcie\_channel\_ffe\_sim

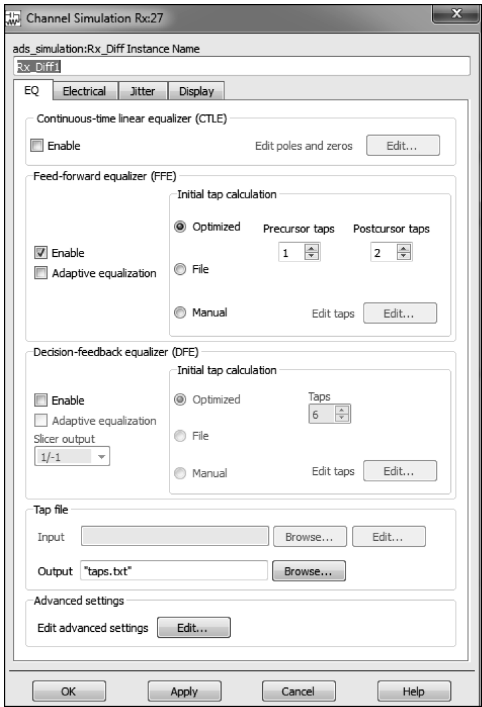


图 7-29 FFE 均衡的设置

- (2) 运行仿真，经过均衡后的眼图、眼高、眼宽如图 7-30 所示。  
(3) 打开工程路径下 data 文件夹中的 taps.txt 文件，优化得到的抽头系数如图 7-31 所示。

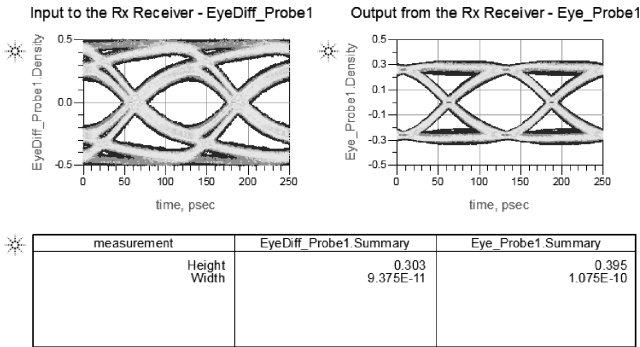


图 7-30 经过 FFE 均衡后的结果

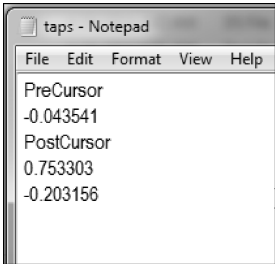


图 7-31 优化得到的抽头系数文件



## 7.6 DFE 均衡

(1) 打开原理图 f\_pcie\_channel\_dfe\_sim，这一设计与前一设计相比除了 Rx\_Diff 部分之外没有任何不同。如图 7-32 所示，打开 Rx\_Diff 设置窗口，在 EQ 标签栏下，Decision - feedback equalizer 部分选择 Enable，设置初始的抽头系数计算方式为 optimized 并设置 Taps = 3。在 Tap file 部分，设置输出抽头系数文件（Output）到 “taps\_dfe.txt”。

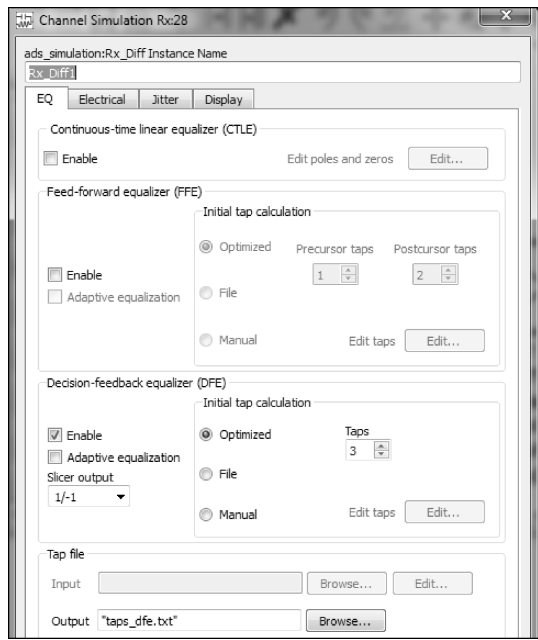


图 7-32 DFE 均衡的设置

(2) 运行仿真，经过均衡后的眼图、眼高、眼宽如图 7-33 所示。

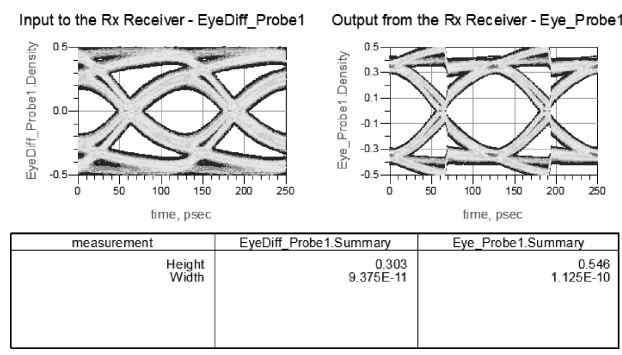



图 7-33 经过 DFE 均衡后的眼图



## 7.7 通道的阶跃响应与冲击响应

(1) 打开原理图 e\_pcie\_channel\_ffe\_sim。选择工具条的 Insert Wire /Pin Label 命令 ，

将 Rx\_diff1 元件的 2 个输入节点分别命名为 Rx\_p、Rx\_n，如图 7-34 所示。

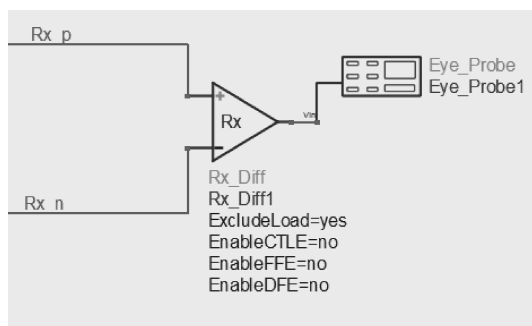


图 7-34 原理图 e\_pcie\_channel\_ffsim

(2) 双击打开通道仿真控制器，选择 Convolution 标签栏下的 Advanced，在打开的对话框中，选中 Save Characterization result，如图 7-35 所示。

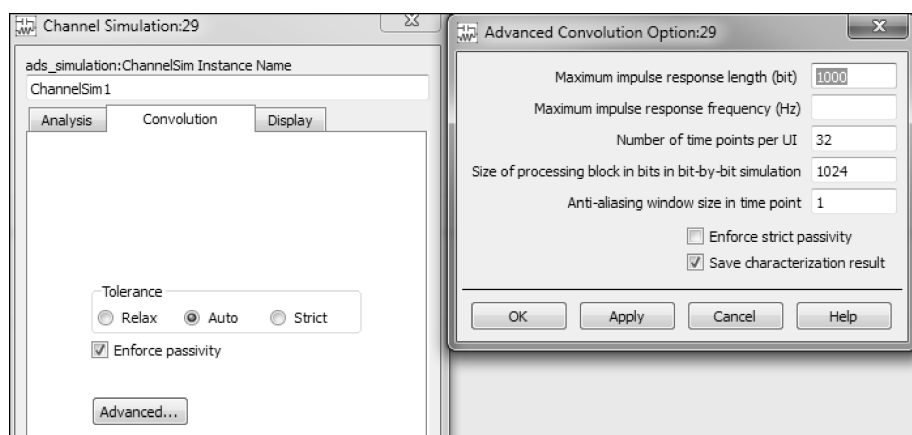


图 7-35 通道仿真器设置

(3) 运行仿真。仿真完成后打开数据显示文件 e\_pcie\_channel\_ffsim. dds，查看仿真的阶跃响应与脉冲响应，如图 7-36 所示。

Calculate the Impulse Response by differentiating the step

Eqn Impulse\_Response=diff(Rx\_p)

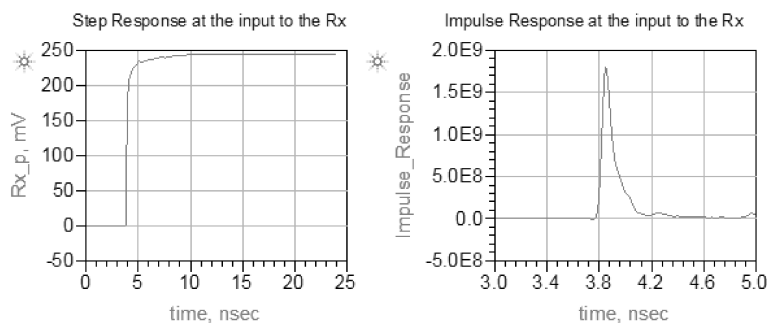


图 7-36 通道的阶跃响应与脉冲响应



## 7.8 IBIS – AMI 模型仿真

(1) 打开原理图 h\_pcie\_ibis\_ami\_bit\_sim, 双击 Tx\_AMI1 发射端元件, 查看 AMI 标签栏下的保留参数 Init\_Returns\_Impulse 和 GetWave\_Exists。这里 Init\_Returns\_Impulse = True, GetWave\_Exists = False, 说明 Tx\_AMI 模型既可用于统计模式仿真, 也可用于逐比特模式仿真, 如图 7-37 所示。

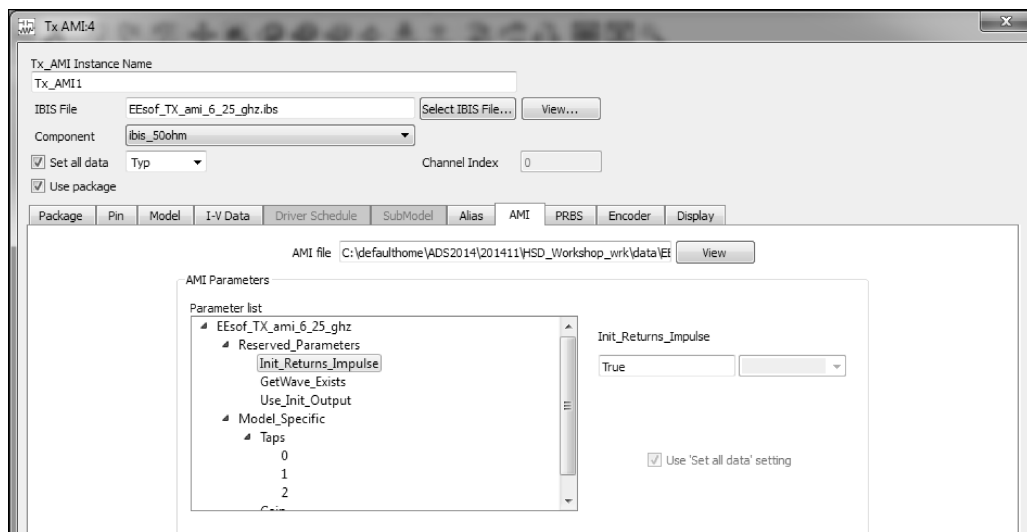
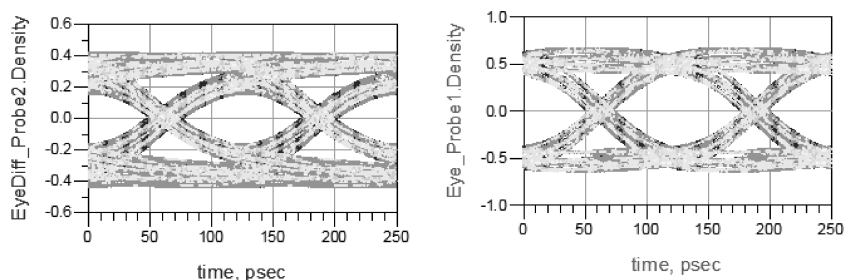


图 7-37 Tx\_AMI 元件的设置

(2) 同样, 查看 Rx\_AMI1 元件下的 Init\_Returns\_Impulse 和 GetWave\_Exists, 确认它也可以用于两种模式的仿真。

(3) 双击通道仿真控制器, 确认仿真模式选择为逐比特模式, 运行仿真。

(4) 仿真结果如图 7-38 所示。



measurement		EyeDiff_Probe2.Summary	Eye_Probe1.Summary
	Height	0.292	0.681
	Width	9.625E-11	1.056E-10

图 7-38 IBIS\_AMI 仿真结果

(5) 打开原理图 i\_pcie\_ibis\_ami\_sta\_sim，确认仿真模式选择为统计模式，运行仿真。仿真结果如图 7-39 所示。

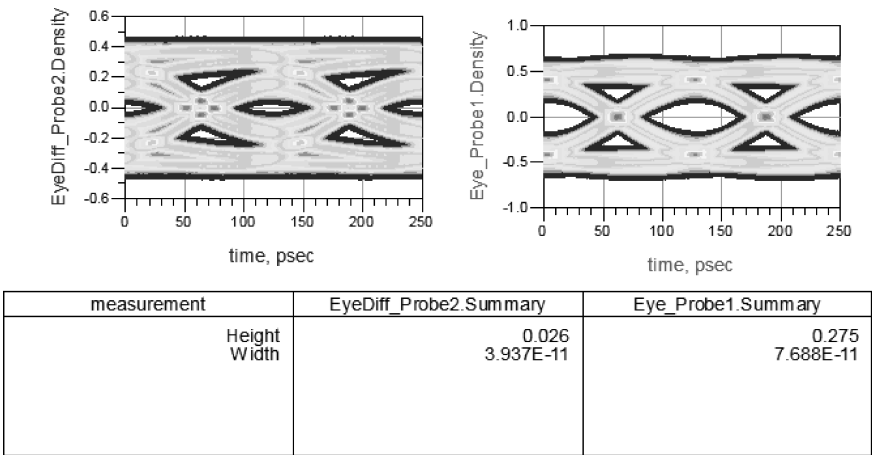


图 7-39 仿真结果



## 7.9 通道参数扫描与最优化

(1) 打开原理图 j\_Tx\_Rx\_batch\_sweep4，如图 7-40 所示。这是一个简单的 AMI 通道仿真实例。下面将通过扫描 AMI 模型参数选择最优的抽头系数组合，以达到眼图最高的目的。

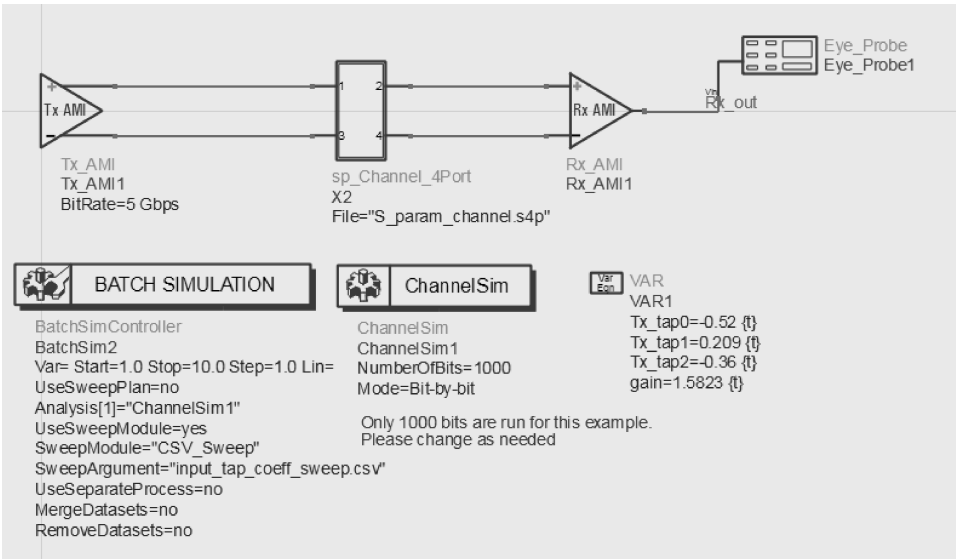


图 7-40 原理图 j\_Tx\_Rx\_batch\_sweep4

(2) 双击 Tx\_AMI1 元件，将 AMI 标签栏下的用户自定义参数“0”、“1”、“2”分别设置为变量 Tx\_tap0、Tx\_tap1、Tx\_tap2，如图 7-41 所示。

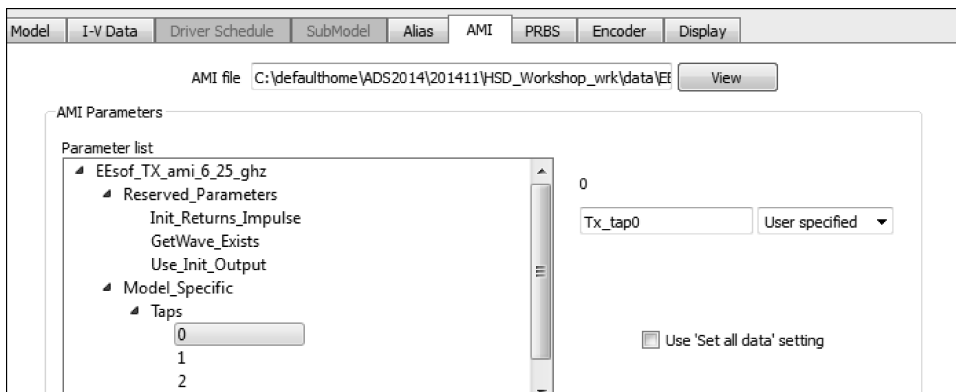


图 7-41 AMI 参数设置

(3) 双击 Batch Simulation 仿真控制器，在 Sweep 标签栏下选择 Use sweep module，Module name 选择 CSV\_Sweep，File name 指定为 “input\_tap\_coeff\_sweep.csv”，如图 7-42 所示。

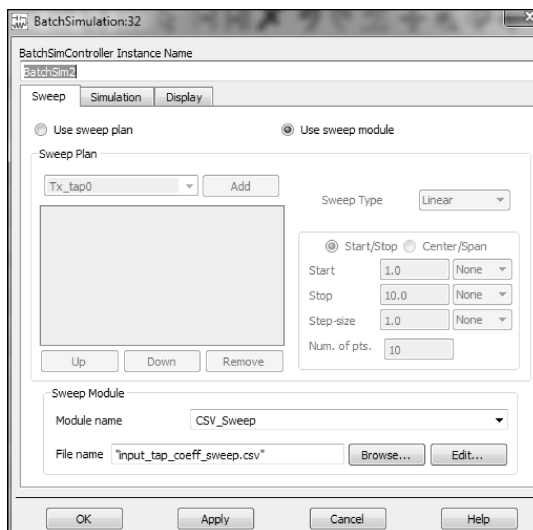


图 7-42 Batch Simulation 仿真控制器

(4) 打开工程路径下 data 文件夹中的 input\_tap\_coeff\_sweep.csv 文件。

文件的第一行：

Tx\_tap0    -0.1    -0.9    0.4 linear

表示扫描变量 Tx\_tap0，扫描范围为 -0.1 ~ -0.9，扫描步进为 0.4，扫描方式为线性扫描，如图 7-43 所示。

	A	B	C	D	E	F	G	H	I
1	Tx_tap0	-0.1	-0.9	0.4	linear				
2	Tx_tap1	0.1	0.8	0.35	linear				
3	Tx_tap2	-0.1	-0.9	0.4	linear				

图 7-43 参数扫描文件

(5) 运行仿真。仿真完成后打开数据显示文件。如图 7-44 所示，图中移动名为 BEST 的 marker 至 Height 最大的位置。此时显示的 BatchNumber = 10 表明第 10 次仿真的结果最优。

图 7-45 显示了第 10 次仿真结果对应的眼图、浴盆曲线及误码率轮廓线。

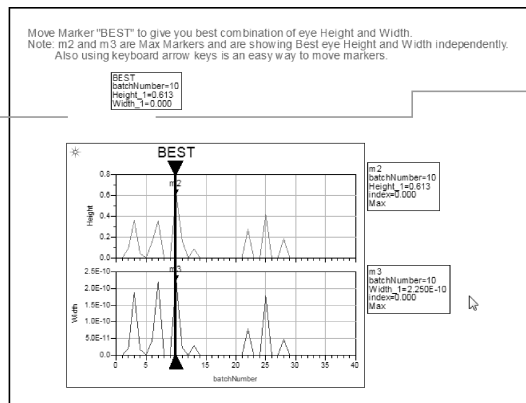


图 7-44 扫描得到的最优组合

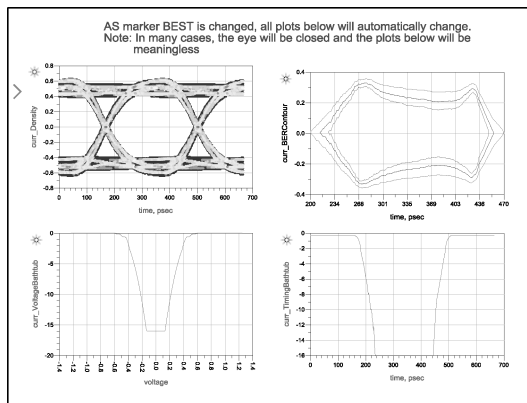


图 7-45 最优组合的仿真结果：  
眼图、浴盆曲线及误码率轮廓线

图 7-46 用列表显示了 batch simulation 扫描的参数值及对应的眼高、眼宽、SNR 等结果。

AS marker BEST is changed, the FIRST ROW (shaded) in the tabular data below will automatically change. The table below shows, inputs like Batch number and swept tap coefficients. And also output like Height, Width and SNR.  
Note: You may also want to scroll through the entire list of tabular data by selecting the list, right clicking on it and choosing Scroll Data.

10.000	-0.100	1.150	-0.100	0.613	2.250E-10	8.706
batchNumber	Tx_tap0	Tx_tap1	Tx_tap2	Height_1	Width_1	SNR_1
1	-0.100	0.100	-0.100	0.000	0.000	1.606
2	-0.100	0.100	-0.500	0.092	2.000E-11	3.463
3	-0.100	0.100	-0.900	0.361	1.883E-10	6.194
4	-0.100	0.450	-0.100	0.041	1.667E-11	3.210
5	-0.100	0.450	-0.500	0.000	0.000	1.144
6	-0.100	0.450	-0.900	0.155	4.500E-11	1.952
7	-0.100	0.800	-0.100	0.353	2.183E-10	5.923
8	-0.100	0.800	-0.500	0.000	0.000	1.637
9	-0.100	0.800	-0.900	0.000	0.000	0.000
10	-0.100	1.150	-0.100	0.613	2.250E-10	8.706
11	-0.100	1.150	-0.500	0.167	2.333E-11	2.378
12	-0.100	1.150	-0.900	0.000	0.000	1.268
13	-0.500	0.100	-0.100	0.086	2.833E-11	3.582
14	-0.500	0.100	-0.500	0.000	0.000	1.116
15	-0.500	0.100	-0.900	0.000	0.000	1.755
16	-0.500	0.450	-0.100	0.000	0.000	0.000
17	-0.500	0.450	-0.500	0.000	0.000	1.690
18	-0.500	0.450	-0.900	0.000	0.000	1.399
19	-0.500	0.800	-0.100	0.000	0.000	1.556
20	-0.500	0.800	-0.500	0.000	0.000	1.554
21	-0.500	0.800	-0.900	0.000	0.000	1.553
22	-0.500	1.150	-0.100	0.273	7.833E-11	2.238
23	-0.500	1.150	-0.500	0.000	0.000	1.632
24	-0.500	1.150	-0.900	0.000	0.000	1.456
25	-0.900	0.100	-0.100	0.415	1.767E-10	6.482
26	-0.900	0.100	-0.500	0.000	0.000	1.721
27	-0.900	0.100	-0.900	0.000	0.000	1.053
28	-0.900	0.450	-0.100	0.182	4.667E-11	2.094
29	-0.900	0.450	-0.500	0.000	0.000	1.383
30	-0.900	0.450	-0.900	0.000	0.000	0.000
31	-0.900	0.800	-0.100	0.000	0.000	1.123
32	-0.900	0.800	-0.500	0.000	0.000	1.541
33	-0.900	0.800	-0.900	0.000	0.000	1.699
34	-0.900	1.150	-0.100	0.000	0.000	0.000

图 7-46 扫描结果列表

### 【总结】

本实例介绍了通道仿真技术的基本原理、仿真流程（包括 IBIS - AMI）和基本设置以及 PCIe3.0 的通道仿真实例。

# 实例 8 使用 IBIS – AMI 模型分析高速串行链路实测波形

## 【目的】

本实例主要介绍如何将高速实时示波器测量的波形结合实际芯片的 IBIS – AMI 模型进行接收的均衡和后处理，分析芯片内眼图和信号质量，拓展高速示波器分析验证的范围，提高设计验证的能力。

通过本实例学习，用户将涉及的操作方法包括：

- ☺ 将示波器波形导入 ADS 进行分析的方法。
- ☺ 使用商用的 IBIS – AMI 模型进行仿真分析的方法。
- ☺ 使用 ADS 进行参数扫描。

## 【背景知识】

### 1. 实时示波器

示波器是一种功能强大的电子测量仪表，能够捕获测量点上的电压波形，观察电子信号的实际情况。尤其在部署新兴的高速串行总线技术、识别宽带宽射频信号的频谱分量或分析瞬时物理现象时，能够得到被测信号的最真实特征。实时示波器有时也称为“单次”示波器，它在每个触发事件上捕获一个完整的波形。也就是说，它在一个连续记录中捕获大量的数据点。

在本案列中测试使用的仪表是 DSAX93204A Infiniium 高性能示波器，支持高达 32GHz 的模拟带宽，具体的介绍请参考网站：<http://www.keysight.com/zh-CN/pd-1819784-pn-DSAX93204A/infiniium-high-performance-oscilloscope-33-ghz?nid=-33821.931874&cc=CN&lc=chi>。

### 2. IBIS – AMI 模型

高速串行链路逐渐成为传输高速数字数据的主要方式。而 IBIS – AMI 是专门针对高速串行链路仿真而创建的一种建模标准，用于对高速串行链路进行快速仿真建模。相对于传统的基于 HSPICE 模型或是 IBIS 模型的仿真，基于 IBIS – AMI 模型的仿真具有以下优点：

- ☺ 仿真速度快，可以在几分钟仿真上百万比特的数据波形。
- ☺ 能够快速仿真分析串扰的影响。
- ☺ 能够快速地进行抖动的仿真分析。
- ☺ 能够仿真分析数据波形序列对最终系统误码率的影响。



☺ 通过引入算法模块支持信号分析的混合建模，包括相应的预加重、均衡、时钟恢复等模块。

通常，完整的 IBIS - AMI 模型包括三种模型文件，即 IBIS 文件（.ibs）、AMI 参数文件（.ami）和算法可执行文件（windows 系统下的 .dll 或是 Linux 系统下的 .os）。

1) **IBIS 文件** 是可读的 ASCII 码文件，并在关键字 [Algorithmic Model] 中声明对应的 AMI 参数文件和算法可执行文件。IBIS 文件中常包括以下内容：

☺ 不包含均衡的发射机特性：输出电压摆幅、输出阻抗、转换速率、输出寄生参数等。

☺ 不包含均衡的接收机特性：输入负载阻抗、寄生参数等。

2) **AMI 参数文件** 也是可读的 ASCII 码文件，通常包含两个主要部分：

☺ [Reserved\_Parameters] 关键字中定义模型的处理流程，比方使用 Init\_Returns\_Impulse 和 GetWave\_Exists 处理通道响应或是使用 Tx\_Jitter、Rx\_Clock\_PDF 加入抖动参数等。

☺ [Model\_Specific] 关键字中把仿真器的参数传递给相应的可执行模块。

3) **算法可执行文件** 描述了信号处理的行为级模型，例如：

☺ 发射芯片的去加重算法实现模块。

☺ 接收芯片的 CTLE（连续时间线性均衡器）、DFE（判决反馈均衡器）、CDR（时钟恢复）模块等。



## 8.1 测量波形

### 1. 测试环境

搭建如图 8-1 的测试环境。

☺ 被测件是高速串行链路通道，包含高速数字收发芯片和 PCB 无源通路。

☺ 测量所用的仪表是德科技 DSAX93204A Infiniium 高性能示波器，支持高达 32GHz 的模拟带宽。

☺ 使用 N2836A 的示波器焊接前端探头和 N2803A 的示波器前端驱动放大器。

在靠近接收芯片的位置的焊盘焊接上 N2836A 示波器探头（见图 8-2），准备开始测量波形数据。

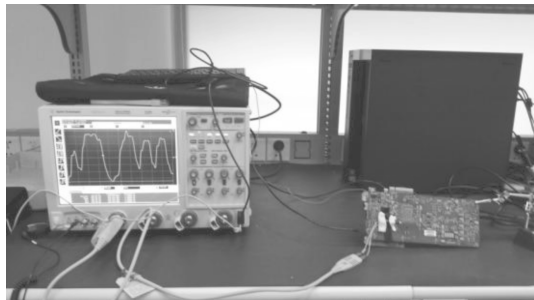


图 8-1 测试环境

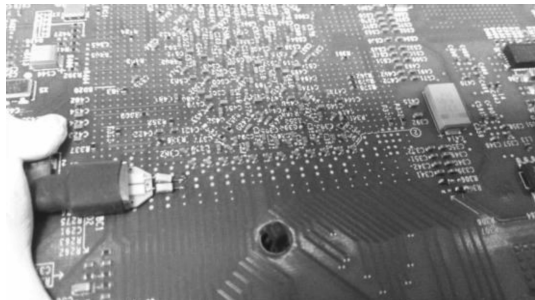


图 8-2 焊接的示波器探头

## 2. 测量波形

被测件上电，并通过软件配置运行在 10Gb/s 的工作状态，并配置发射芯片的预加重参数。通过示波器观察波形和眼图，如图 8-3 所示。运行示波器软件菜单 Setup→Acquisition，弹出如图 8-4 所示的窗口，并按照图 8-4 中的参数进行设置。

- ☺ Sampling Mode: Real Time
- ☺ Sin(x)/x Interpolation: On (4pts)
- ☺ Sampling Rate: 80GSa/s
- ☺ Memory Depth: 2 Mpts

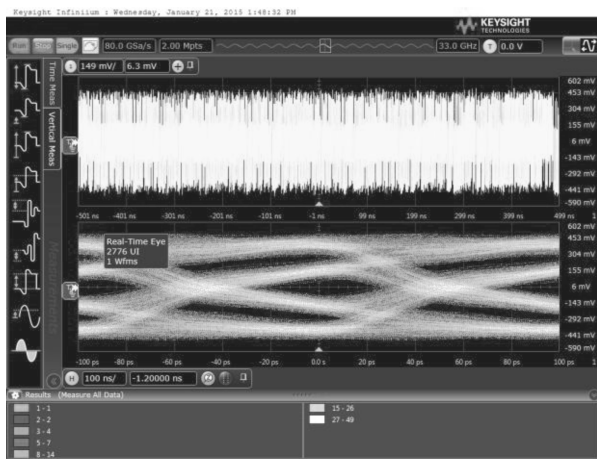


图 8-3 示波器测量界面



图 8-4 示波器捕获设置

经过以上设置，示波器内部存储器上写入的波形工作在 320GSa/s 的采样率上，并记录了 8M 个采样点。

## 3. 导出波形文件

运行示波器软件菜单 File→Save→Waveform..., 弹出如图 8-5 的窗口，并按照图 8-5 所示进行设置。单击“Save”，会有一个新文件 redriver\_wvfm1.txt 保存在目录下。

- ☺ File Name: redriver\_wvfm1
- ☺ Save as Type: Waveform Files (Y Values)
- ☺ Waveform Sources Options: all data

使用文本编辑器打开生成的 redriver\_wvfm1.txt 文件，并在文件开头添加一行文字“3.125E-12 8000000.0”，如图 8-6 所示，其中“3.125E-12”是对应于 320GSa/s 的采样时间间隔，“8000000.0”是指 8M 个采样点。

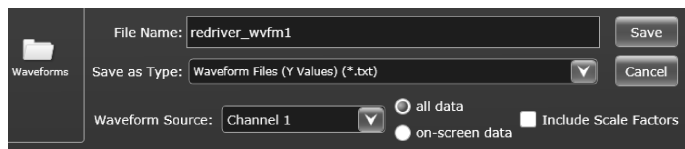


图 8-5 保存波形文件

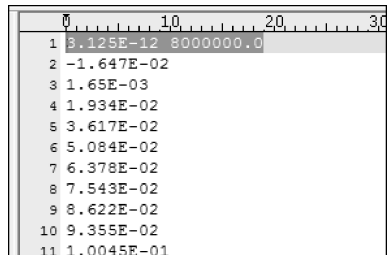


图 8-6 编辑波形

保存修改后的文件，至此后续步骤中所需要的实测波形文件已经准备就绪，该文件 (redriver\_wvfm1.txt) 可在实例下载资源中查找。



## 8.2 通道建模

### 1. 下载芯片模型

打开 Keysight ADS 软件，新建工程文件 wvfm\_AMI\_wrk，假设工程文件所在目录为 \wvfm\_AMI\_wrk。

进入 Altera 官方网站，下载 Stratix V GX/GS 的 IBIS - AMI 模型，下载地址是 <http://www.altera.com/download/board-layout-test/hspice/hsp-index.html>，下载的文件名是 stratix5gx\_hssi\_ibis\_ami\_models\_v2p7.zip，如图 8-7 所示。

Cyclone V GX/GT	5CGX 5CGT	Transceiver —IBIS-AMI	<a href="#">cyclone5axgt_hssi_ibis_ami_models_v1.2.zip</a>	1.2	12/2014
Stratix V GX/GS	5SGX, 5SGS	Transceiver —Synopsys HSPICE	<a href="#">stratix5gx_hssi_hspice_models_v1p0.zip</a>	1.0	05/2014
		Transceiver —IBIS-AMI	<a href="#">stratix5gx_hssi_ibis_ami_models_v2p7.zip</a>	2.7	08/2014
Stratix V GT	5SGT	Transceiver —Synopsys HSPICE	<a href="#">s5gt_EAP_hssi_hspice_model_v0p1.zip</a>	1.0	07/2013

图 8-7 IBIS - AMI 模型下载地址

将压缩文件中与 rx 有关的文件 (见图 8-8) 放入 wvfm\_AMI\_wrk\data 文件夹中，包括 .ibs/.ami/.dll 文件。

再将压缩文件中与封装有关的 S 参数文件 (见图 8-9) 放入 wvfm\_AMI\_wrk\data 文件夹中，即 svgx\_rx\_pkg.s4p。

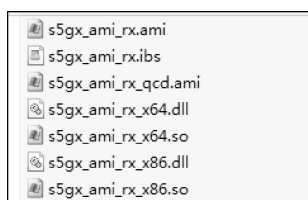


图 8-8 IBIS - AMI 文件

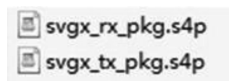


图 8-9 封装 S 参数文件

### 2. PCB 通道建模

由于示波器测试点无法点在 BGA 的管脚上，因此测试点距离芯片的 ball 还有一小段 PCB 链路，可以通过 ADS 的电磁场仿真完成局部 PCB 通道的建模。

解压附件中 ADS 工程文件 PCB\_wrk。打开工程文件中 a10\_fpga\_with\_Slot\_adfi 的 layout 设计，如图 8-10 所示。

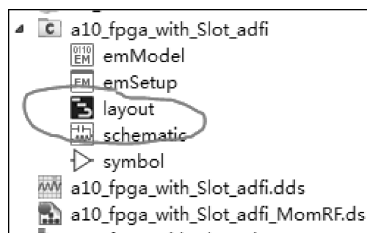


图 8-10 layout 工程文件

这个 layout 是原始的 Allegro PCB 版图，通过 ADS 的 ADFI 流程截取局部电路直接得到（见图 8-11）。通过导入生成的 .adfi 文件，自动生成的 layout 设计已经包含了相应的端口、层叠定义和材料属性，具体过程这里不再赘述。接下来是相应的电磁场设定仿真过程。

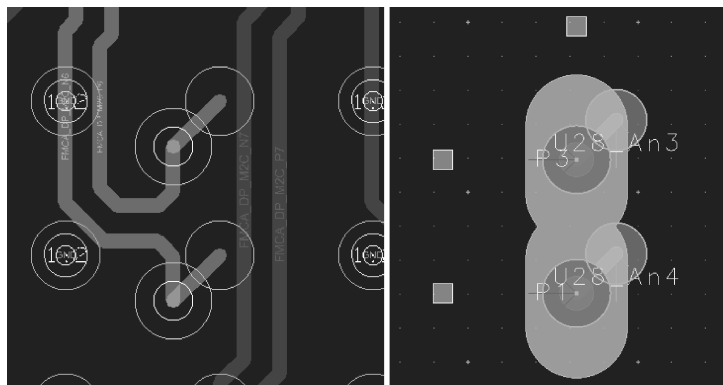


图 8-11 仿真版图

在 Layout 窗口中单击“3D View”的快捷图标（见图 8-12），弹出 3D 浏览窗口（见图 8-13）。在 3D 浏览窗口中可以看出两对端口分别在 Top 层和 Bottom 层的四个圆形焊盘上。

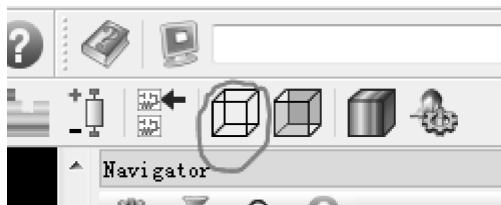


图 8-12 3D View 快捷图标

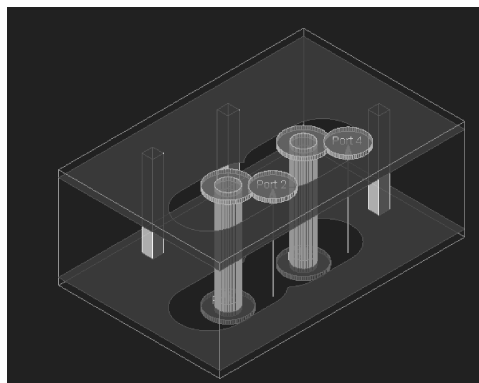


图 8-13 3D 浏览窗口

接下来设置面端口。在 layout 窗口，单击菜单 Options→Layer Preferences...，在弹出的窗口中，单击“hide all”，之后只选中“ETCH\_BOTTOM”这一层，如图 8-14 所示。

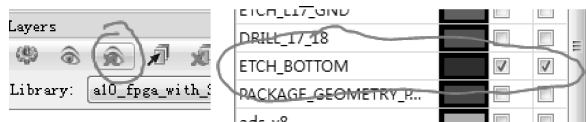


图 8-14 Layer Preferences 选项

选中下方的端口和圆形焊盘，单击菜单 Edit→Edge/Area Pin...，在弹出窗口中的下拉菜单中选择“P1(1)”，单击 OK 按钮，如图 8-15 所示。端口 1 的面端口设置就完成了。以此类推，把剩下的三个端口设置成面端口。

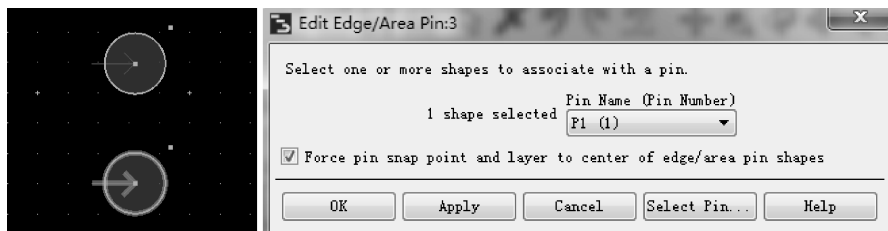


图 8-15 Area Pin 的设置

在 layout 窗口，单击菜单 EM→Simulation Setup...，在弹出窗口中首先选择使用的电磁仿真器（见图 8-16）。

☺ Setup Type: EM Simulation/Model

☺ EM Simulator: Momentum RF

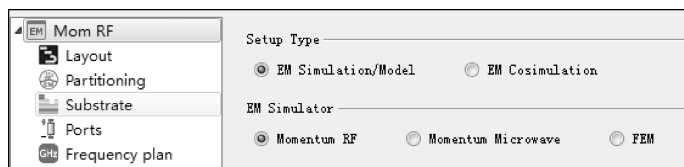


图 8-16 仿真器的选择

左边的菜单栏切换到“Substrate”，在右边的下拉菜单中选择“a10\_fpga\_with\_Slot\_cut”，如图 8-17 所示。

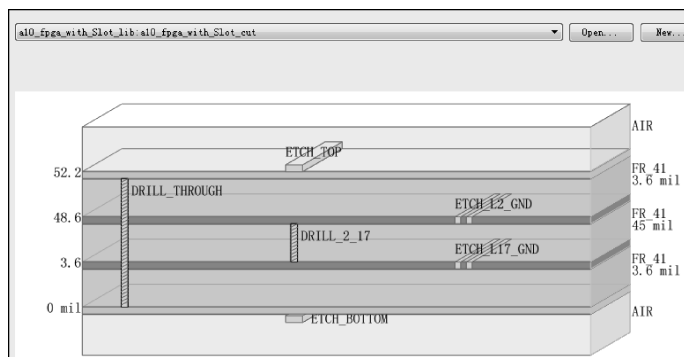


图 8-17 层叠的设置

左边的菜单栏切换到“Frequency Plan”，如图 8-18 进行设置。

☺ Type: Linear

☺ Fstop: 30GHz

☺ Fstart: 0GHz

☺ Step: 0.1 GHz

左边的菜单栏切换到“Model”，如图 8-19 所示勾选“Auto - update EM Model when simulation is launched”。

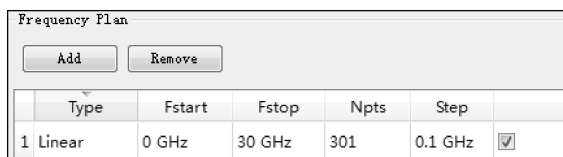


图 8-18 设置频率扫描

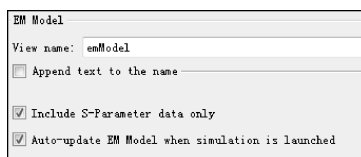


图 8-19 EM Model 的设置

单击右下角的“simulate”，仿真 S 参数。经过若干分钟的仿真，在 Display 窗口弹出仿真的 S 参数结果，如图 8-20 所示。建立“Symbol”，如图 8-21 所示。

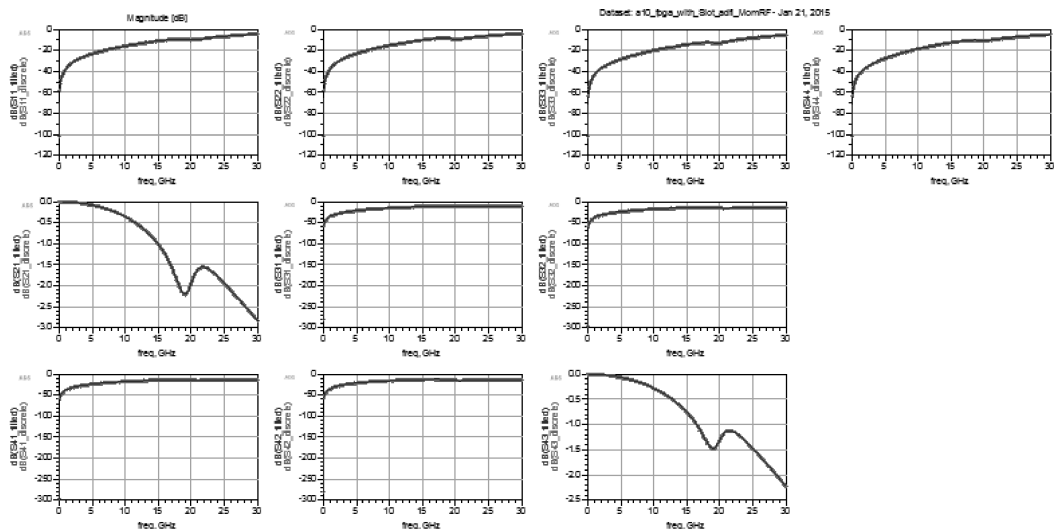


图 8-20 S 参数仿真结果

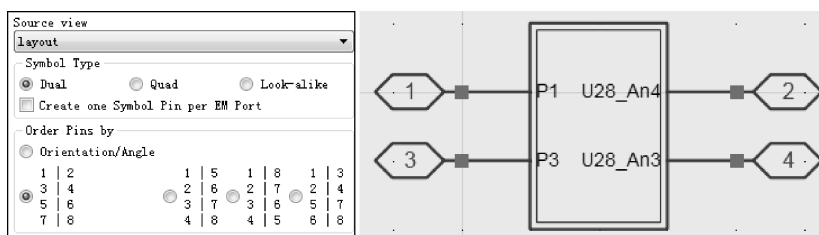


图 8-21 建立“Symbol”



## 8.3 模型验证

### 1. 验证芯片的阻抗模型

打开之前建立的工程文件 wvfm\_AMI\_wrk，新建原理图 Z\_IBIS。在原理图中添加左侧菜单栏 Signal Integrity - IBIS 中的 IBIS 模块，如图 8-22 所示。

双击 IBIS 模块，按照图 8-23 所示选择 IBIS 模型和相应的模型，单击 OK 按钮确定。

☺ IBIS File: s5gx\_ami\_rx.ibs      ☺ Component: stratix\_v\_gx\_rx

☺ Use package: 不勾选

☺ Pin: 20 stratix5\_gx\_rx\_P stratix5\_gx\_rx

☺ Model Selector: stratix5\_gx\_rx\_100 100

按照图 8-24 编辑原理图，并编辑下列模块和仿真器的参数。

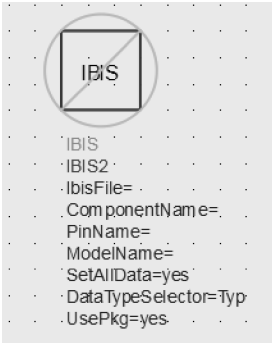


图 8-22 IBIS 模块

IBIS File	3D_Rick_v1_wrk\data\IBIS_AMI\stratix5gx_rx.ibs	Select IBIS File...	View...
Component	stratix_v_gx_rx		
<input checked="" type="checkbox"/> Set all data	Typ		
<input type="checkbox"/> Use package			
Package	Pin	Model	I-V Data   Driver Schedule   SubModel   Alias   Display
Name	Signal Name	Model Name	
20	stratix5_gx_rx_P	stratix5_gx_rx	
21	stratix5_gx_rx_N	stratix5_gx_rx	
Model Selector			
Model	Description		
stratix5_gx_rx_qcd	For use with SiSoft's QCD, only!		
stratix5_gx_rx_100	100		
stratix5_gx_rx_85	85		

图 8-23 IBIS 模型的选择

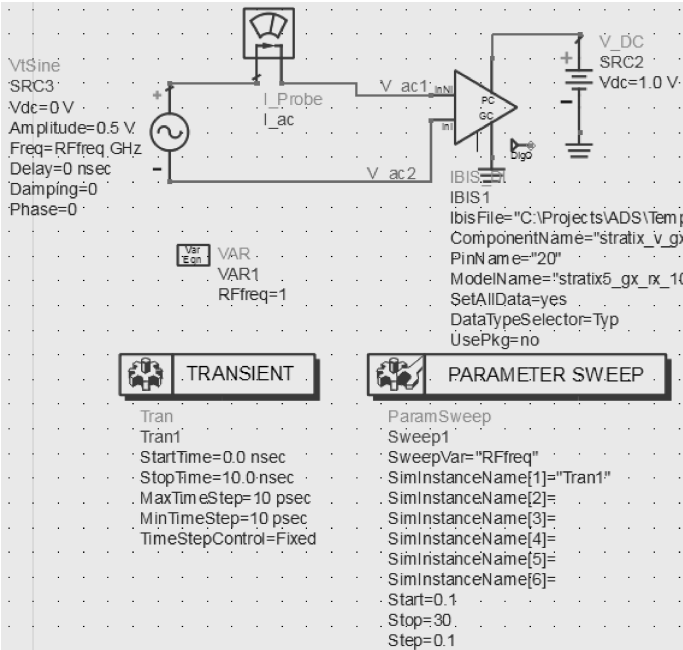


图 8-24 原理图 Z\_IBIS

VAR:

RFfreq = 1

VtSine:

Freq = RFfreqGHz

Amplitude = 0.5 V

I\_Probe:

Instance Name: I\_ac

V\_DC:

Vdc = 1.0 V

Tran(Transient 仿真器):

StartTime = 0.0 nsec

StopTime = 10.0 nsec

```
MaxTimeStep = 10 psec
MinTimeStep = 10 psec
TimeStepControl = Fixed
ParmSweep
SweepVar = " RFfreq"
Start = 0. 1
Stop = 30
Step = 0. 1
SimInstanceName[ 1 ] = " Tran1 "
```

按照图 8-24 连线和添加 “GND” 器件，并且在 IBIS 模型的输入端口的连线上使用快捷图标 “NAME”（见图 8-25）添加电压节点：V\_ac1 和 V\_ac2。单击快捷图标 “Simulate”（见图 8-25）运行仿真。

仿真结束弹出显示窗口，使用窗口左侧菜单栏的快捷图标 “Equation” 添加公式，如图 8-26 所示。



图 8-25 快捷图标

```
EqnV_f=fs(V_ac1-V_ac2)
EqnI_f=fs(I_ac.i)
EqnZin=max(V_f)/max(I_f)
```

图 8-26 编辑公式

使用窗口左侧菜单栏的快捷图标 “list” 添加参数 “Zin” 的列表（见图 8-27）、使用快捷图标 “Rectangular Plot” 添加图表（见图 8-28），输入如下等式，显示输入阻抗随频率变化的关系。

```
plot_vs( mag( Zin) , RFfreq)
plot_vs( real( Zin) , RFfreq)
plot_vs( mag( imag( Zin) ) , RFfreq)
```

RFfreq	Zin
0.100	101.872 -j0.543
0.200	101.864 -j1.087
0.300	101.849 -j1.630
0.400	101.829 -j2.173
0.500	101.803 -j2.715
0.600	101.771 -j3.257
0.700	101.733 -j3.799
0.800	101.690 -j4.340
0.900	101.641 -j4.880
1.000	101.586 -j5.420
1.100	101.525 -j5.959
1.200	101.459 -j6.496
1.300	101.387 -j7.033
1.400	101.309 -j7.569
1.500	101.226 -j8.104
1.600	101.137 -j8.638
1.700	101.042 -j9.170

图 8-27 阻抗列表

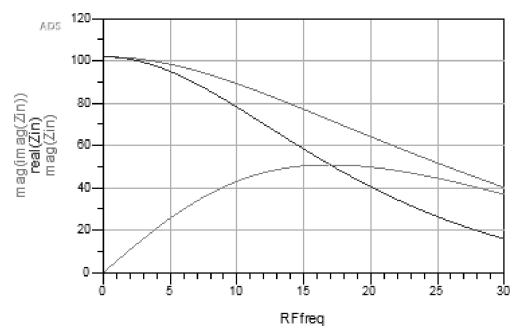


图 8-28 阻抗图表

从图 8-28 中可以看出，低频时接收芯片的 Die 表现出约 100Ohm 的差分输入阻抗，随着频率的升高，实阻抗逐渐减小，虚阻抗逐渐增大，这也与模型的物理特性相一致。

## 2. 直通波形验证

打开工程文件 wvfm\_AMI\_wrk，新建原理图 meas\_Step1。在原理图窗口，单击菜单栏 DesignGuide→IBIS AMI→Sample AMI Models→Tx Pass Through 和 Rx Pass Through，如图 8-29



所示。单击后, 文件“EEsof\_Tx\_Pass\_Through. ibs”和“EEsof\_Rx\_Pass\_Through. ibs”会出现在 wvfm\_AMI. wrk\data 文件夹中。

从左侧菜单栏中找到 Signal Integrity→Tx AMI/Rx AMI 两个模块并拖入原理图, 如图 8-30ADS 中 AMI 模块。

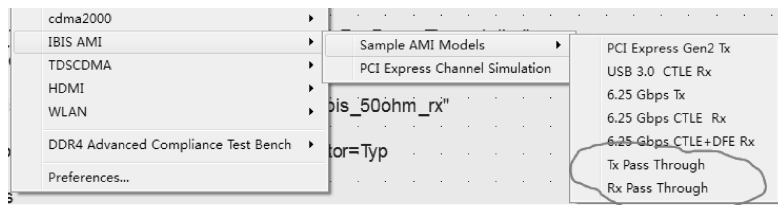


图 8-29 直通模型

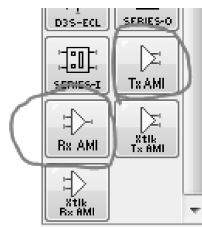


图 8-30 ADS 中 AMI 模块

按照图 8-31 编辑原理图, 修改编辑下列模块和仿真器的参数, 并确认波形文件 redriver\_wvfm1. txt 已保存在 wvfm\_AMI. wrk\data 文件夹中。

VAR

\_\_debug\_InputRedriver = 1

ChannelSim

NumberOfBits = 100000

Mode = Bit - by - bit

Tx\_AMI

IbisFile = "EEsof\_Tx\_Pass\_Through. ibs"

BitRate = 10 Gbps

Rx\_AMI

IbisFile = "EEsof\_Rx\_Pass\_Through. ibs"

Eye\_Probe

Measurement Selection: Density is selected

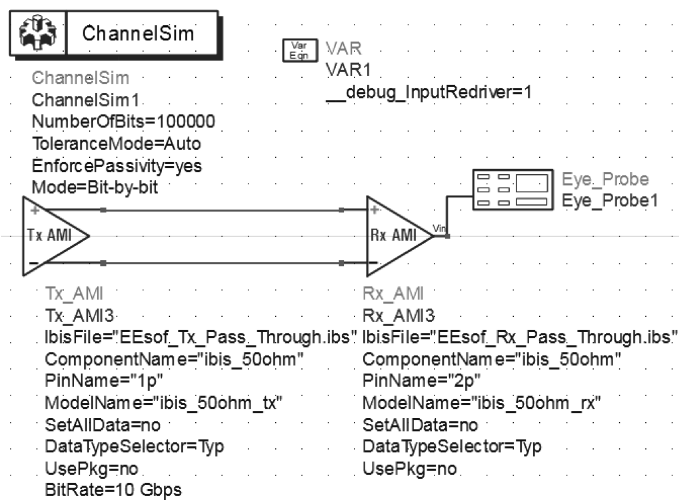


图 8-31 原理图 meas\_Step1

单击“Simulate”运行仿真，在弹出的显示窗口添加“Rectangular Plot”，显示“Density”，如图 8-32 所示。图 8-33 所示的是显示的眼图结果，与之前用示波器测试的眼图保持一致。

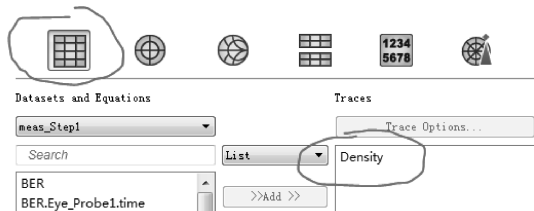


图 8-32 添加眼图

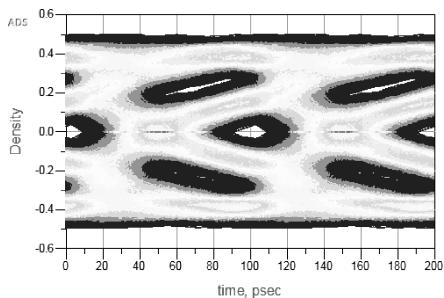


图 8-33 仿真眼图结果

### 3. 无源通道验证

打开工程文件 wvfm\_AMI\_wrk，确认 wvfm\_AMI\_wrk\data 文件夹中包含 svgx\_rx\_pkg.s4p，用文本编辑器打开文件，确认端口定义（见图 8-34）。

新建原理图 Altera\_Pkg，按照图 8-35 编辑原理图，修改编辑下列模块，并使用快捷图标“Insert Pin”添加端口。

S4P

File = "svgx\_rx\_pkg.s4p"

Type = Touchstone

```
! Port definitions:
! 1) Bump_N
! 2) Bump_P
! 3) Ball_N
! 4) Ball_P
```

图 8-34 S 参数文件中端口定义

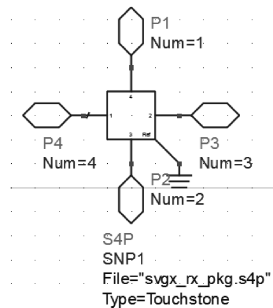


图 8-35 原理图 Altera\_Pkg

为原理图 Altera\_Pkg 建立“Symbol”，如图 8-36 所示。

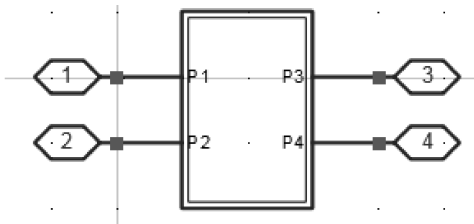
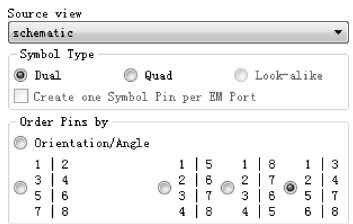


图 8-36 为原理图 Altera\_Pkg 建立 Symbol

在 ADS 的主窗口单击菜单栏 DesignKits→Manage Libraries…。在弹出的窗口中单击“Add Library Definition File…”，之后选择 PCB\_wrk 文件夹下的“lib.defs”文件。可以看到有新的 Library 加载到工程文件中，如图 8-37 所示。

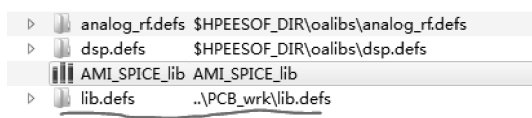


图 8-37 加载 Library

新建原理图 Pkg\_sim，按照图 8-38 编辑原理图，修改编辑下列模块和仿真器的参数。Altera\_Pkg 调用原理图，a10\_fpga\_with\_Slot 调用 emModel。

S\_Param

Start = 0GHz

Stop = 30GHz

Step = 0. 10GHz

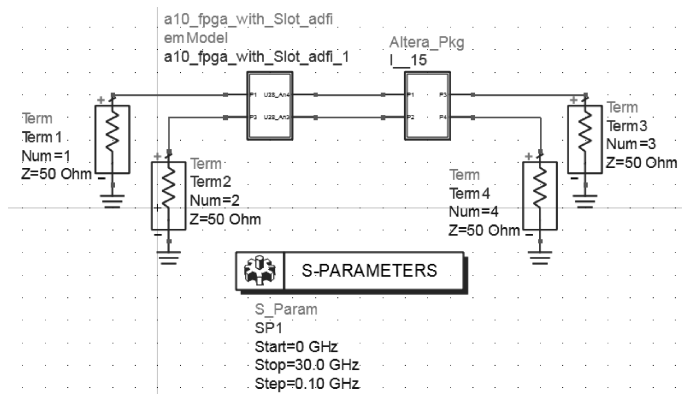


图 8-38 原理图 Pkg\_sim

运行仿真，并在结果窗口中显示 S 参数，如图 8-39 所示。

新建原理图 Pkg\_Model，并建立相应的“Symbol”，如图 8-40 所示。

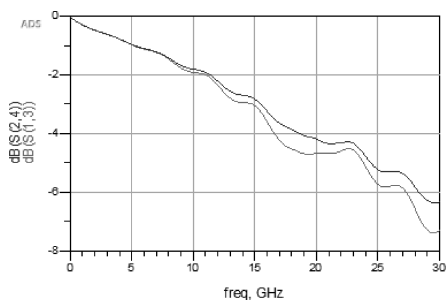


图 8-39 S 参数仿真结果

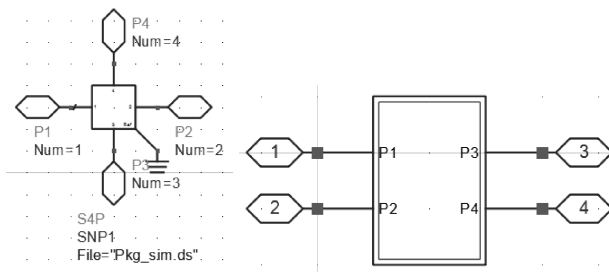


图 8-40 原理图 Pkg\_Model 及其 Symbol



## 8.4 AMI 模型的后处理和分析

打开工程文件 wvfm\_AMI\_wrk，新建原理图 meas\_Step2。按照图 8-41 编辑原理图，修改编辑下列模块和仿真器的参数，并确认在 8.1 节生成的波形文件 redriver\_wvfm1.txt 已保存在 wvfm\_AMI\_wrk\data 文件夹中。

VAR

\_\_debug\_InputRedriver = 1

ChannelSim

NumberOfBits = 100000

Mode = Bit - by - bit

EnforcePassivity = yes

Tx\_AMI

IbisFile = "EEsof\_Tx\_Pass\_Through. ibs"

BitRate = 10 Gbps

Rx\_AMI

IBIS File = "s5gx\_ami\_rx. ibs"

ComponentName = "stratix\_v\_gx\_rx"

PinName = "20"

ModelName = "stratix5\_v\_gx\_rx\_100"

SetAllData = no

Use package = no

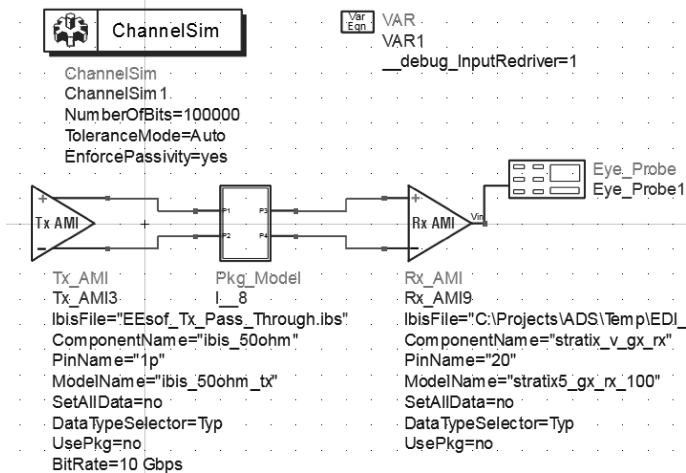


图 8-41 原理图 meas\_Step2

在以上编辑的基础上，双击 Rx\_AMI，修改参数（见图 8-42）。

Model\_Specific

AMI file: s5gx\_ami\_rx. ami

rx\_power: 0

```
rx_bw:      0
per_vccela: 0
rx_dcgain:  0
rx_acgain:  0
corner:     0

dfe
mode:       2

debug
enable:     True
dump_getwave_input: False
dump_getwave_output: False
dump_dfe_adaptation: True
dump_cdr_adaptation: True
```

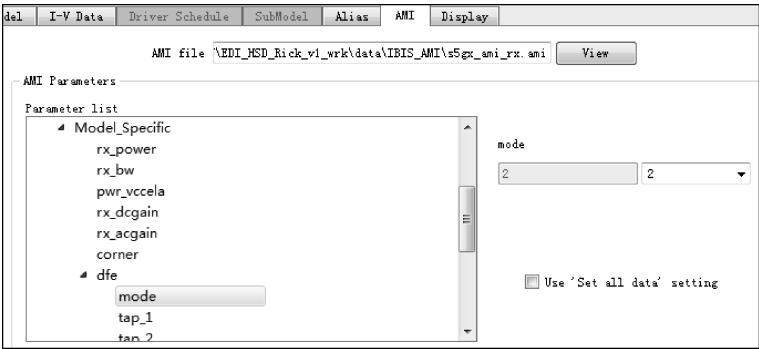


图 8-42 编辑 AMI 模型的参数

单击“Simulate”，运行仿真，在显示窗口中使用“Rectangular Plot”显示“Density”，如图 8-43 所示。可见经过 AMI 模型的处理，眼图和原始波形（见图 8-33）有明显的不同。

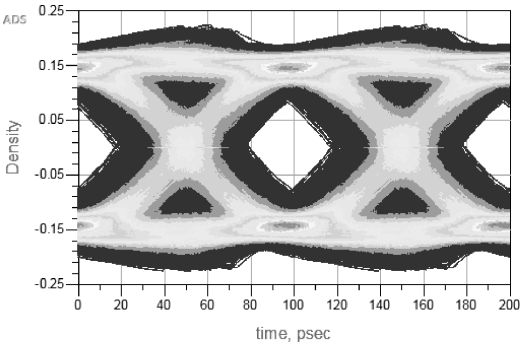


图 8-43 眼图仿真结果

同时打开仿真后生成的文件 wvfm\_AMI.wrkdir\data\Altera\_S5GX\_Rx\_AMI\_Debug.txt，在文件中记录了 AMI 模型输出的一些参数设置，可以作为配置实际芯片的参考，如图 8-44 所示，其中也包括自适应 DFE 均衡器的参数。

Tap 1	Tap 2	Tap 3	Tap 4	Tap 5
1	-1	1	-3	1

```
At time, 1.500000e-005 s, input jump: 0.002820 V.
DFE input: max. = 0.201106, min. = -0.196240.
DFE output: max. = 0.211620, min. = -0.217680.
CDR is locked at time, 1.500000e-005 s.
At time, 1.500000e-005 s, output jump: 0.002432 V.
(Stratix5_Rx (Description "Altera AmiRxFFT derivative.") (tap_1 1) (tap_2 -1) (tap_3 1) (tap_4 -3) (tap_5 1) )
```

图 8-44 输出报告



## 8.5 扫描参数

通过 8.4 节可以得到 DFE 均衡器参数及其对应的眼图。但是芯片的 CTLE 的参数需要手动配置。可以通过 ADS 的参数扫描，寻找优化的设置。

新建原理图 meas\_Step2\_sweep，拷贝图 8-41 中的所有模块到新的原理图中，编辑 Rx\_AMI 模块的 AMI 选项，如图 8-45 所示。

```
Model_Specific
    AMI file:      s5gx_ami_rx. ami
    rx_power:      0
    rx_bw:         0
    per_vccela:    0
    rx_dcgain:     dc_gain( user defined)
    rx_acgain:     ac_gain( user defined)
    corner:        0
dfe
    mode:          2
debug
    enable:        False
```



图 8-45 编辑 AMI 模型参数

同时添加一些模块到原理图中，如图 8-46 所示。运行仿真，参数 ac\_gain 会对值 0/3/6/9 进行扫描，参数 dc\_gain 会对值 0/1/2 进行扫描。

```
Var
    dc_gain = 0
    ac_gain = 0
BatchSimController
```

Var = "ac\_gain" Start = 0 Stop = 9 Step = 3 Lin =

Var = "dc\_gain" Start = 0 Stop = 2 Step = 1 Lin =

UseSweepPlan = yes

Analysis[1] = "ChannelSim1"

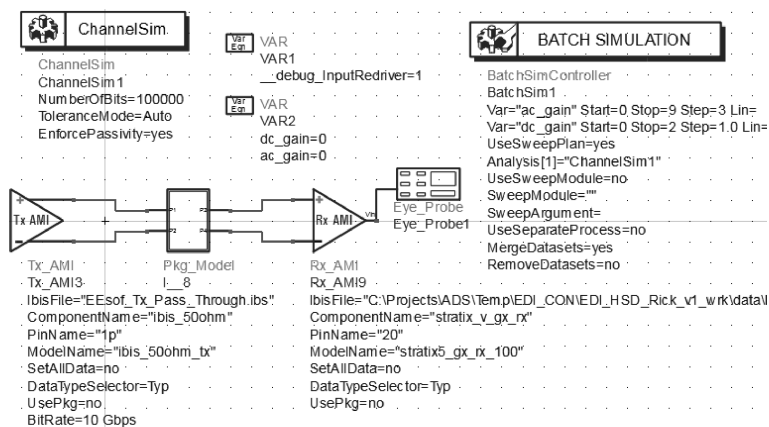


图 8-46 原理图 meas\_Step2\_sweep

在结果窗口可以使用公式  $\text{Density}[\text{index\_ac\_gain}, \text{index\_dc\_gain}, :, :]$  对设置了不同的 CTLE 参数的眼图结果进行显示（见图 8-47），其中  $\text{index\_ac\_gain}$  的范围是 0 ~ 3， $\text{index\_dc\_gain}$  的范围是 0 ~ 2，如  $\text{Density}[1, 2, :, :]$  就表示  $\text{ac\_gain} = 3$ 、 $\text{dc\_gain} = 2$  时的眼图。

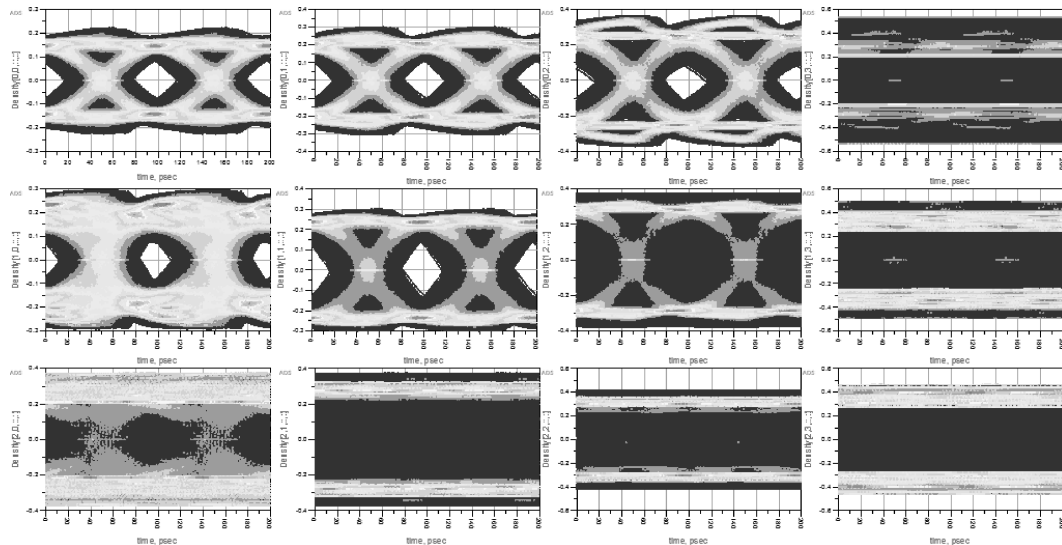


图 8-47 批量仿真眼图结果

同时可以使用“list”选择“Width”和“Height”显示眼高和眼宽，如图 8-48 所示。可以看出当  $\text{dc\_gain} = 1$ ， $\text{ac\_gain} = 3$  时眼图张开幅度较大，这与图 8-47 所示的结论是一致的。

ac_gain	permute(Width)	permute(Height)
index=0.000, dc_gain=0.000		
0.000	4.650E-11	0.191
3.000	3.400E-11	0.154
6.000	0.000	0.114
9.000	0.000	0.000
index=0.000, dc_gain=1.000		
0.000	4.500E-11	0.255
3.000	4.450E-11	0.350
6.000	0.000	0.291
9.000	0.000	0.000
index=0.000, dc_gain=2.000		
0.000	4.550E-11	0.270
3.000	2.500E-11	0.338
6.000	1.100E-11	0.358
9.000	0.000	0.000
index=0.000, dc_gain=3.000		
0.000	1.400E-11	0.051
3.000	1.100E-11	0.319
6.000	0.000	0.357
9.000	0.000	0.000

图 8-48 批量仿真眼宽眼高结果

【总结】

本实例主要介绍如何将高速实时示波器测量的波形结合实际芯片的 IBIS - AMI 模型进行接收的均衡和后处理，分析眼图和信号质量，拓展高速示波器分析验证的范围。同时通过仿真，可以学习瞬态仿真、通道仿真的设置方法，了解 IBIS - AMI 模型的参数设置以及参数扫描及优选的方法，全方位提高针对高速信号完整性问题设计验证的能力。



# 实例 9 光通道 IBIS – AMI 模型创建及验证

## 【目的】

通过本实例，在 SystemVue 中进行典型光通道 IBIS – AMI 模型的创建，并使用 ADS 通道仿真器对其进行验证。

## 【背景知识】

### 1. IBIS – AMI 模型介绍

IBIS – AMI 模型的相关介绍请参考实例 8，在此不再赘述。

### 2. 光通道介绍

光收发模块在 25Gbps 以上的高速互联结构中集成光电技术有着非常大的潜力，其带宽大、功耗低、体积小等特点使得其应用越来越广泛。

图 9-1 所示为光收发模块的典型架构。

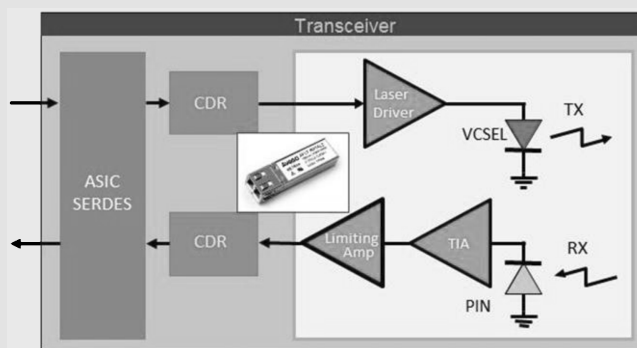


图 9-1 光收发模块典型架构

光收发模块主要包括激光驱动器、垂直腔面发射激光器（VCSEL）、光检波器、跨阻放大器（TIA）、限幅放大器、光纤以及时钟恢复模块（CDR）。光收发模块通过可插拔接头和主板进行电气连接。

在高速数据通信中，如 FFE、CTLE、DEF 等均衡技术以及时钟恢复技术被大规模应用以提高数字信号质量。必须对光收发模块进行建模，将其和通道的其他部分进行整体考虑，以预测整个通道性能。

对于光收发模块供应商而言，由于没有 AMI 建模经验，提供合格的 IBIS – AMI 模型需要耗费数月的时间。信号完整性工程师及 IC 设计师对于 Matlab/C++ 等编程工具并不熟悉，也不了解 AMI 规则，故开发 AMI 模型对光收发模块供应商而言难度很大。



## 9.1 在 SystemVue 中进行光通道 IBIS – AMI 模型创建

进行光通道 IBIS – AMI 模型创建需要使用到的软件平台为 SystemVue2013.08SP1；Visual Studio C++ 2010 SP1 Express Edition 进行 IBIS – AMI 模型编译；使用 FlexDCA 进行波形验证；使用 ADS2014.01 进行包含 IBIS – AMI 模型的整个通道仿真。

在示例工程中，首先针对光模块中的关键器件垂直腔面发射激光器（VCSEL）进行建模及验证，再对整个光模块进行建模验证，最后生成 IBIS – AMI 模型，导入 ADS 进行通道仿真验证。

将 SystemVue 安装目录下 Example 文件夹中的 OpticalAMI.wsv 工程文件及测试数据 measurement\_data\_device\_1.txt 复制至当前目录中，以进行 IBIS – AMI 模型创建实验（见图 9-2）。

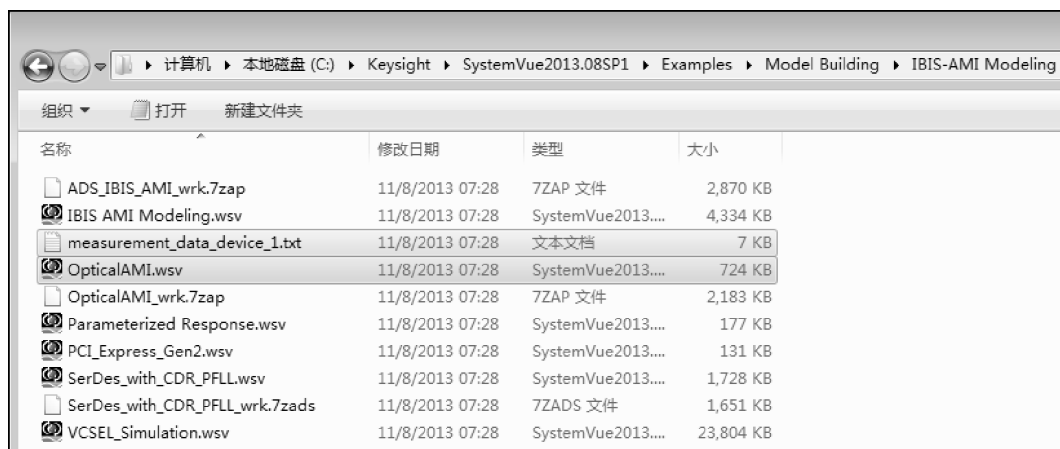


图 9-2 SystemVue 案例中光收发模块相关文件

### 1. VCSEL 参数提取

在光通道中，垂直腔面发射激光器是进行电光转换的关键器件。在 SystemVue 中使用 VCSEL 模型来进行实际器件模拟。由于 VCSEL 工作在非常高的速率下，故不太容易直接获取动态行为模型。取而代之的是，读取静态 L – I 及 V – I 测试曲线，进行 VCSEL 行为模型提取，并将提取结果存入数据文件。

打开 OpticalAMI 工程文件中的设计 Design\_VCSEL\_Extraction，进行激光器参数提取（见图 9-3）。

VCSEL\_Extractor 在设计中起到两个作用：对实测曲线 measurement\_data\_device\_1.txt 进行参数提取，存为 Device1.dat，供后续激光器建模使用；将输入端的电流激励信号转化为输出端的电压信号，同时比较原始器件输出信号及参数提取模型输出信号。

图 9-4 所示为对比 VCSEL 模型输出的 L – I 曲线和原始 L – I 曲线。

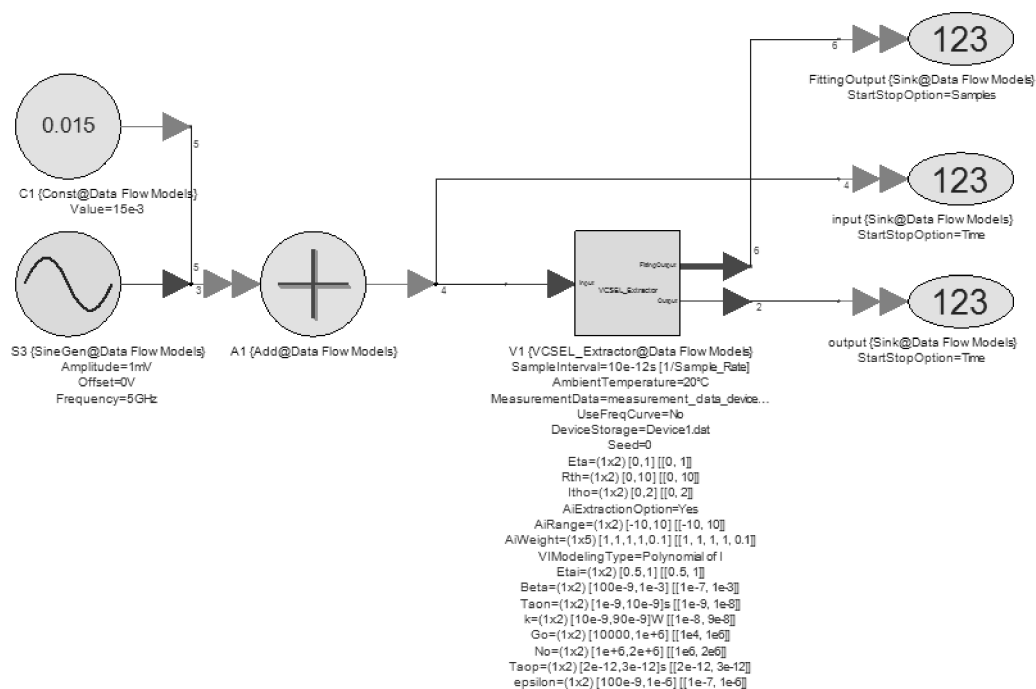


图 9-3 VCSEL 参数提取原理图

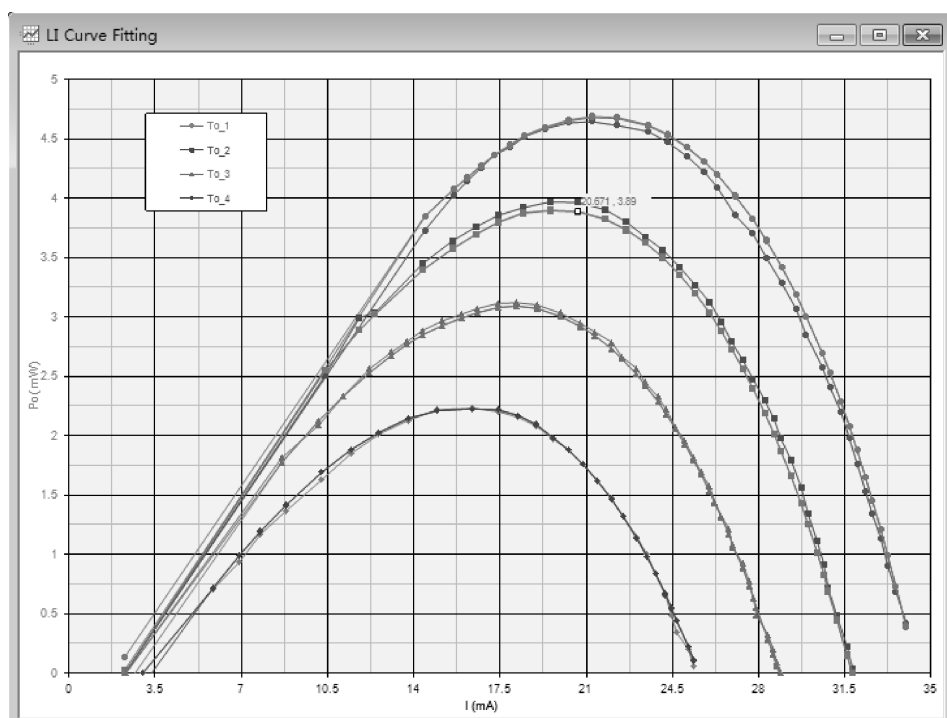


图 9-4 VCSEL 模型输出 LI 曲线对比

## 2. 典型光通道构建

光通道模块包括激光源、光纤、PIN 二极管及跨阻放大器 (TIA) 等器件。在模型 Opti-

calLink 中有完成的光通道模块案例。

在模块的输入端,使用 VCSEL 驱动器来将差分输入信号转换为单端电流信号,如图 9-5 所示。通过设置增益、输出 1dB 压缩点、偏置电流、上升下降时间、抖动及采样间隔等关键指标来描述驱动放大器的特性。在 VCSEL 模型中调用上一步中提取的数据文件 Device1.dat,并设定工作温度等参数,如图 9-6 所示。

随后使用放大器模型来对激光器到光纤的耦合损耗进行建模,如图 9-7 所示。

多模光纤模型包括了光纤的物理模型及衰减系数等指标,如图 9-8 所示。

differential voltage --> single-ended current

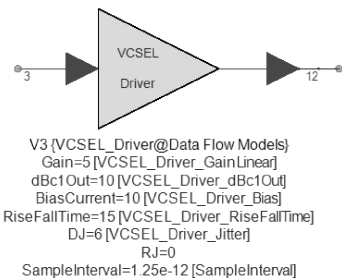


图 9-5 VCSEL 驱动器

single-ended current --> square root of light energy

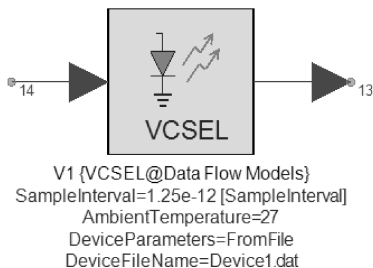


图 9-6 VCSEL 模型

VCSEL to fiber coupling loss

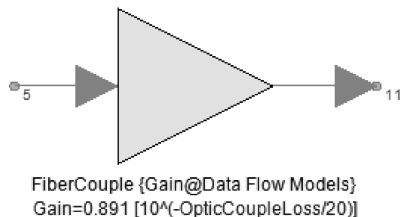


图 9-7 耦合损耗模型

光纤模型后使用光滤波器去除带外噪声,如图 9-9 所示。

multi-mode fiber  
(pulse broadening and attenuation)

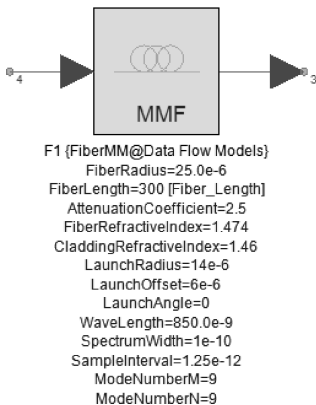


图 9-8 多模光纤模型

optical filter  
(remove out-of-band noise)

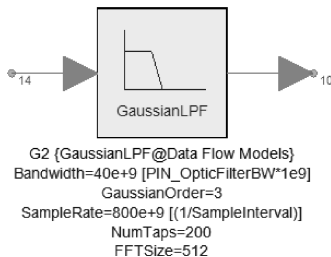


图 9-9 光滤波器模型

使用 PIN 光检波器将光信号转换为电流信号,如图 9-10 所示。使用低通滤波器来滤除带外噪声,如图 9-11 所示。使用 IIR 滤波器来去除信号中的直流分量,如图 9-12 所示。

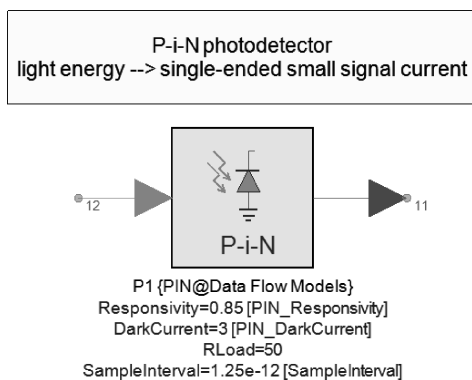


图 9-10 PIN 光检波器模型

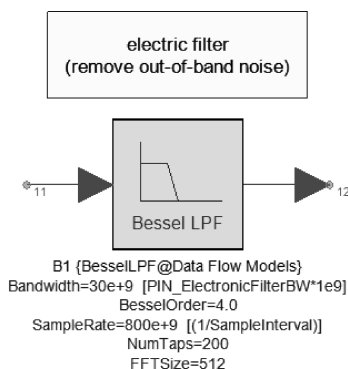


图 9-11 低通滤波器模型

最后使用跨阻放大器将差分电流信号转换为差分电压信号，并进行放大，如图 9-13 所示。

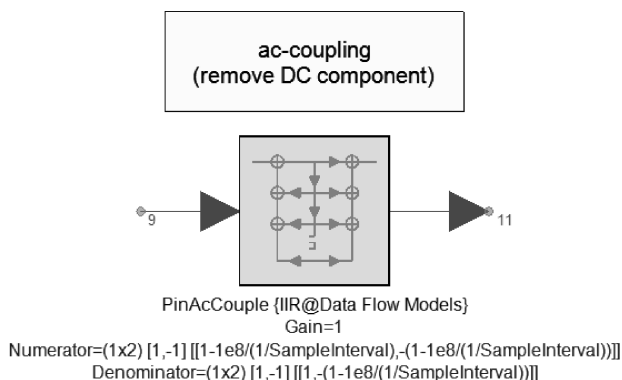


图 9-12 IIR 滤波器

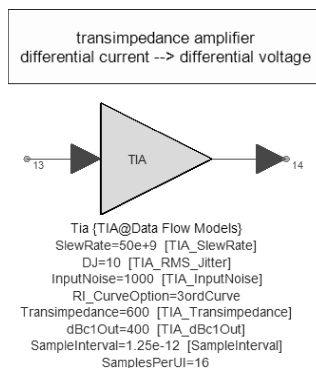


图 9-13 跨阻放大器模型

将上述器件进行级联，并加入输入输出端口。完成后的光通道模块如图 9-14 所示。

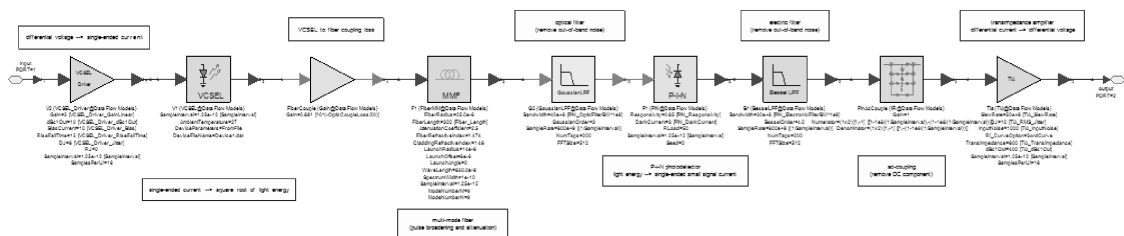


图 9-14 完整光通道模块

在光通道构建完毕后，可以在 SystemVue 中进行仿真，验证其性能。图 9-15 是 Design1 中的测试电路，首先观察捕获到的时域波形。

图 9-16 所示为经过光通道后的时域信号波形。将 Sink 状态改为开路，并将 FlexDCA\_Sink 激活，能够观察经过光通道后的实时信号波形，如图 9-17 所示。使用 FlexDCA 软件，能够对信号的时域波形进行捕获，还能够描绘眼图、进行抖动分析等，如图 9-18 所示。

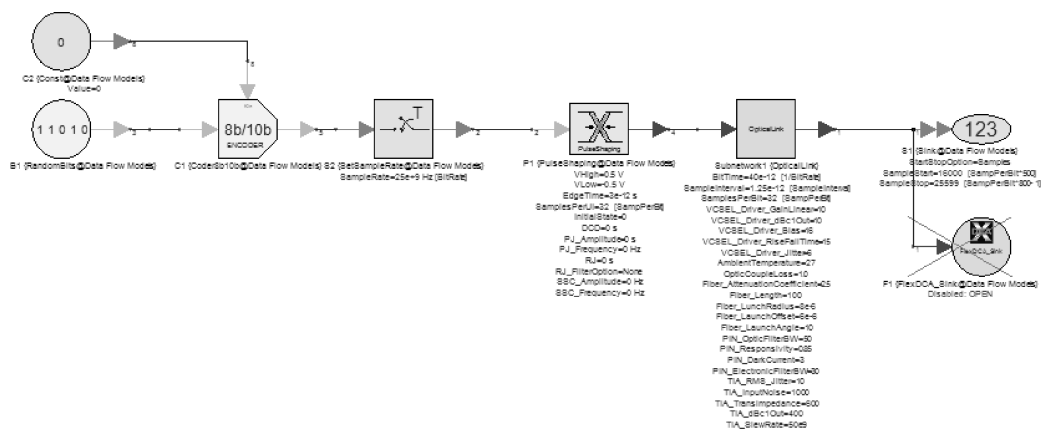


图 9-15 对光通道模块进行验证

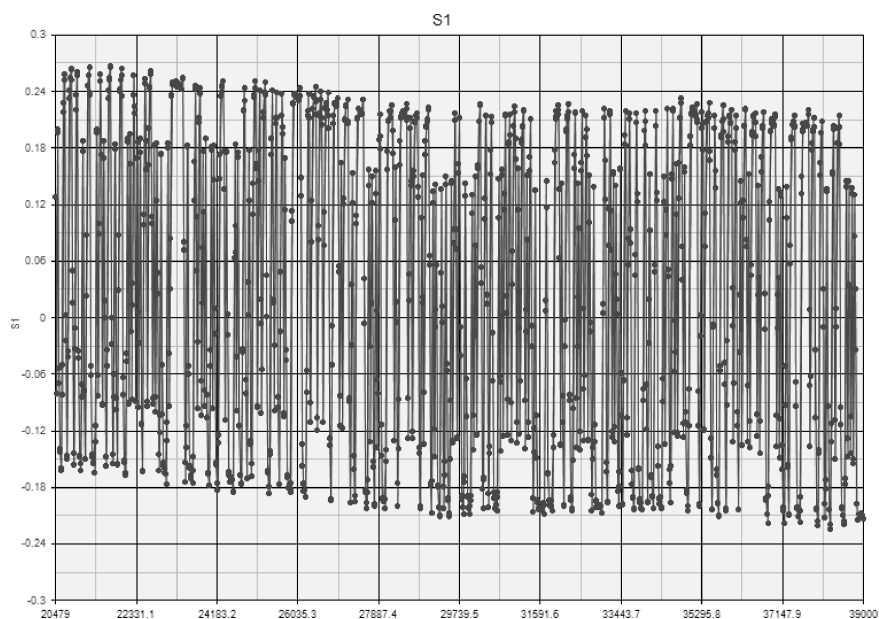


图 9-16 光通道输出时域信号

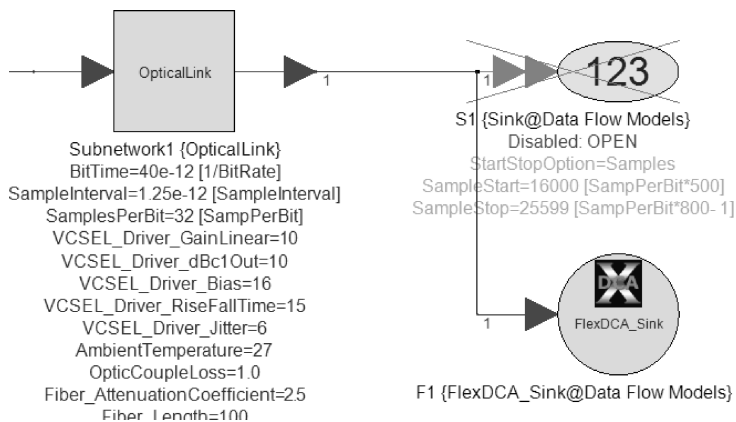


图 9-17 激活 FlexDCA\_Sink

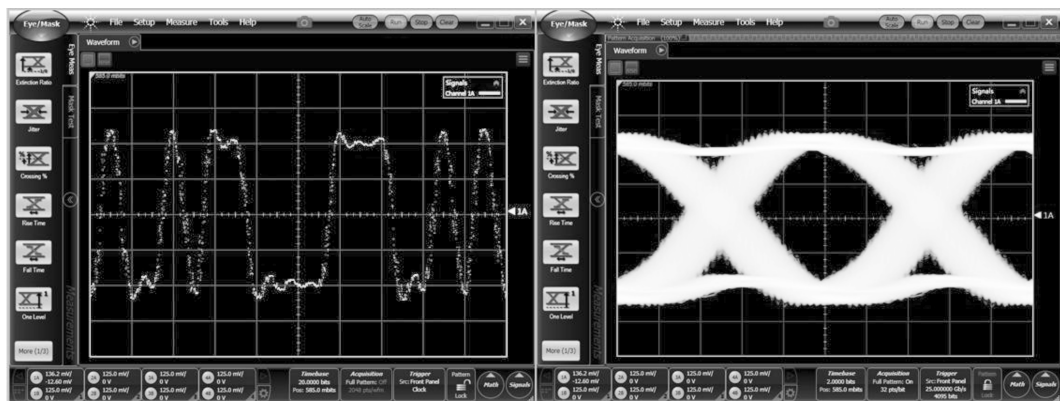


图 9-18 FlexDCA 软件界面

### 3. 光通道 IBIS - AMI 模型生成

光通道模型创建并进行验证后，使用 SystemVue 的 AMI 代码生成功能，能够自动配置 AMI 参数，生成模型。

在目录树上选择工程文件名，添加 Code Generators \ C ++ Code Generator，获得如图 9-19 所示的页面。

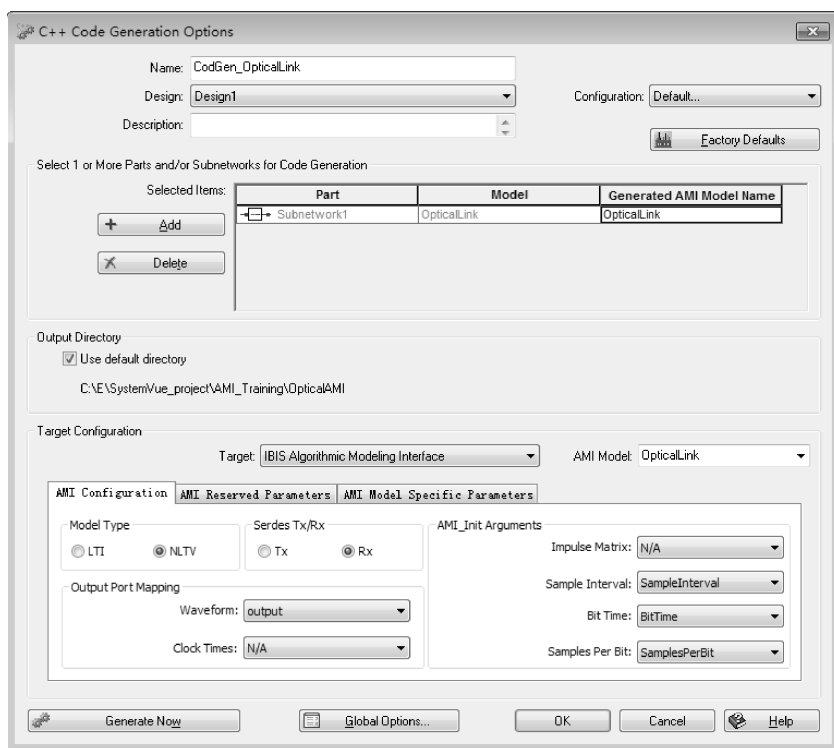


图 9-19 代码生成器界面

在 AMI Model Specific Parameters 页面下，可以选择需要输出的参数（见图 9-20）。

参数设置完毕后，单击 Generate Now，系统将会自动调用 Visual - Studio，生成 .ibs 文件、32 位及 64 位 dll 文件、ami 文件及说明 txt 文档，如图 9-21 所示。

AMI Configuration		AMI Reserved Parameters	AMI Model Specific Parameters
Name	Export	Properties	
VCSEL_Driver_GainLinear	<input checked="" type="checkbox"/>	(VCSEL_Driver_GainLinear (Usage In) (Type Float) (Format Value 10) (Default 10) (Descrip	
VCSEL_Driver_dBc1Out	<input checked="" type="checkbox"/>	(VCSEL_Driver_dBc1Out (Usage In) (Type Float) (Format Value 10) (Default 10) (Description	
VCSEL_Driver_Bias	<input checked="" type="checkbox"/>	(VCSEL_Driver_Bias (Usage In) (Type Float) (Format Value 16) (Default 16) (Description "V	
VCSEL_Driver_RiseFallTim	<input checked="" type="checkbox"/>	(VCSEL_Driver_RiseFallTime (Usage In) (Type Float) (Format Value 15) (Default 15) (Descri	
VCSEL_Driver_Jitter	<input checked="" type="checkbox"/>	(VCSEL_Driver_Jitter (Usage In) (Type Float) (Format Value 6) (Default 6) (Description "VC	
AmbientTemperature	<input checked="" type="checkbox"/>	(AmbientTemperature (Usage In) (Type Float) (Format Value 27) (Default 27) (Description "	
OpticCoupleLoss	<input checked="" type="checkbox"/>	(OpticCoupleLoss (Usage In) (Type Float) (Format Value 1) (Default 1) (Description "OpticC	
Fiber_AttenuationCoefficient	<input checked="" type="checkbox"/>	(Fiber_AttenuationCoefficient (Usage In) (Type Float) (Format Value 2.5) (Default 2.5) (Des	

图 9-20 选择需要输出的 AMI 参数

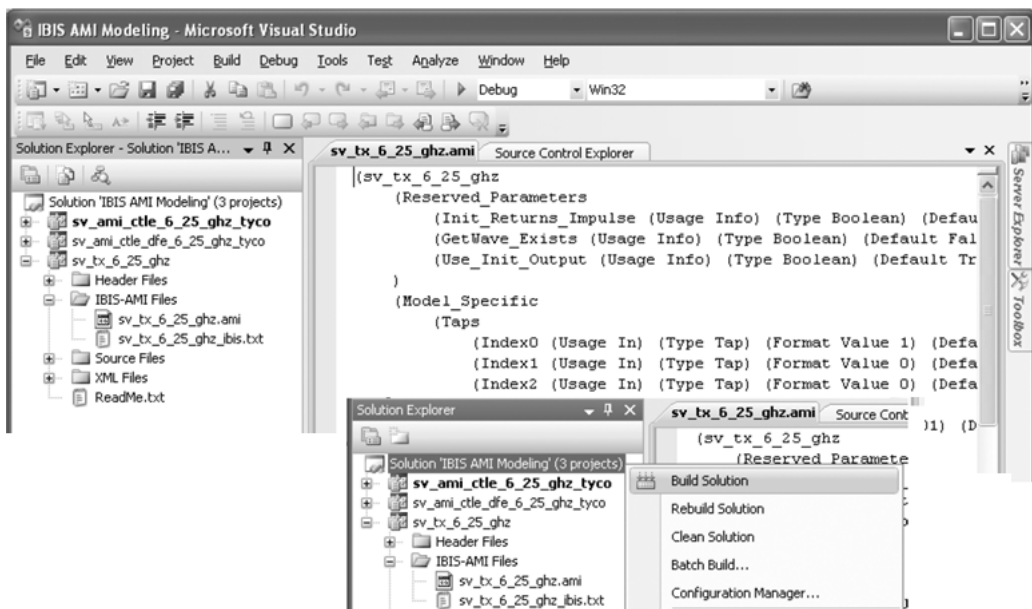


图 9-21 生成 IBIS - AMI 文件

在进行通道仿真时，需要使用 ibs 文件、ami 文件及 dll 文件。



## 9.2 在 ADS 中进行光通道 IBIS - AMI 模型验证

光通道模型可以归类为重驱动 AMI 模型。由于重驱动模型包含两个背对背常规 AMI 模型，其分别代表器件的接收和发射部分，如图 9-22 所示。

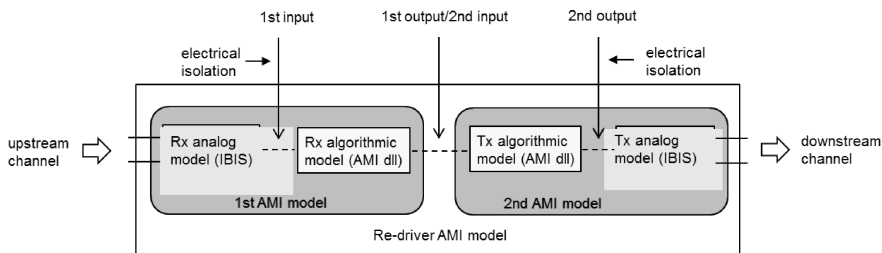


图 9-22 重驱动 AMI 模型示意图



每个部分的模型都包含其 .ami 或 .dll (或 .so) 文件, 但都含有相同的 .ibs 文件。第一级 AMI\_Init 接收前级通道冲击响应来作为输入, 第二级 AMI\_Init 产生冲击响应来作为后级通道的冲击输入。第一级算法模型的输出波形是第二级算法模型的输入信号。故重驱动模型能够在一个通道中进行级联。

在 ADS 中首先针对光通道模块 IBIS - AMI 模型作为输入或输出部分进行验证, 然后再作为重驱动模型进行验证。

1. IBIS - AMI 模型作为发射或接收模型进行验证

将 SystemVue 安装目录下 Example 文件夹中的 OpticalAMI\_wrk.7zap 文件拷贝至 ADS 工作区, 并使用 ADS2014.01 进行解压缩, 对 SystemVue 创建的 IBIS - AMI 模型进行验证。将刚产生的 .ibs 文件、dll 文件、ami 文件拷贝至 ADS 工作区下 data 文件夹中, 覆盖原有的几个文件。在进行 N 级重驱动 IBIS - AMI 模型验证时, 需要逐级验证重驱动模型的正确性。

首先, 调用 ADS 的理想发射 IBIS - AMI 模型作为源, 经过通道模型后, 终端使用重驱动 IBIS - AMI 模型作为接收端, 如图 9-23 所示。

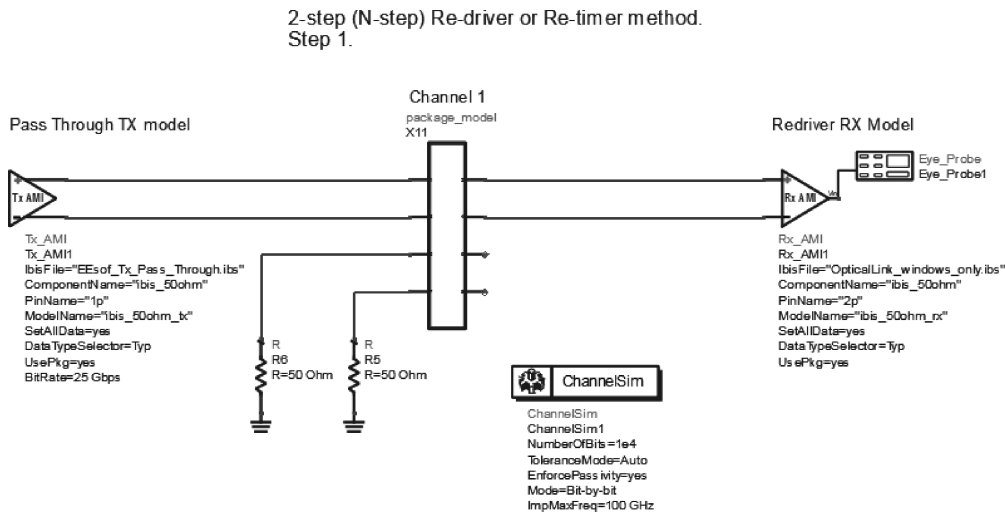


图 9-23 光通道模型作为接收端进行验证

仿真结果如图 9-24 所示。

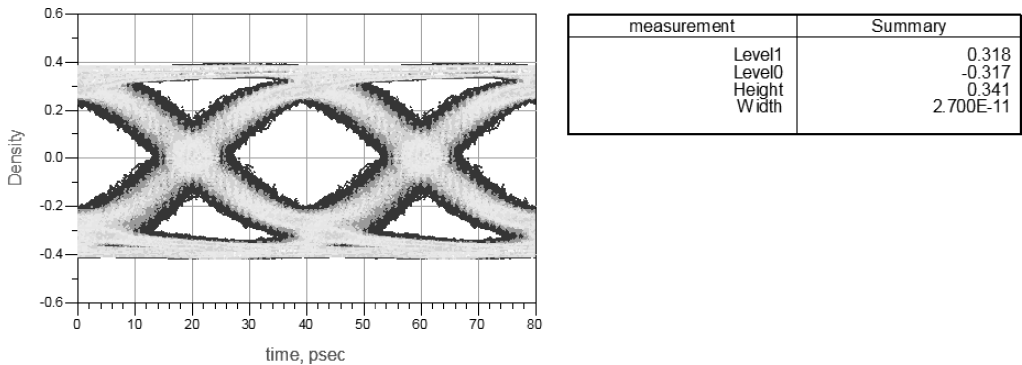


图 9-24 仿真结果

再将重驱动 IBIS - AMI 模型作为发射端，经过通道模型后，端接 ADS 的理想接收 IBIS - AMI 模型作为接收端，如图 9-25 所示。

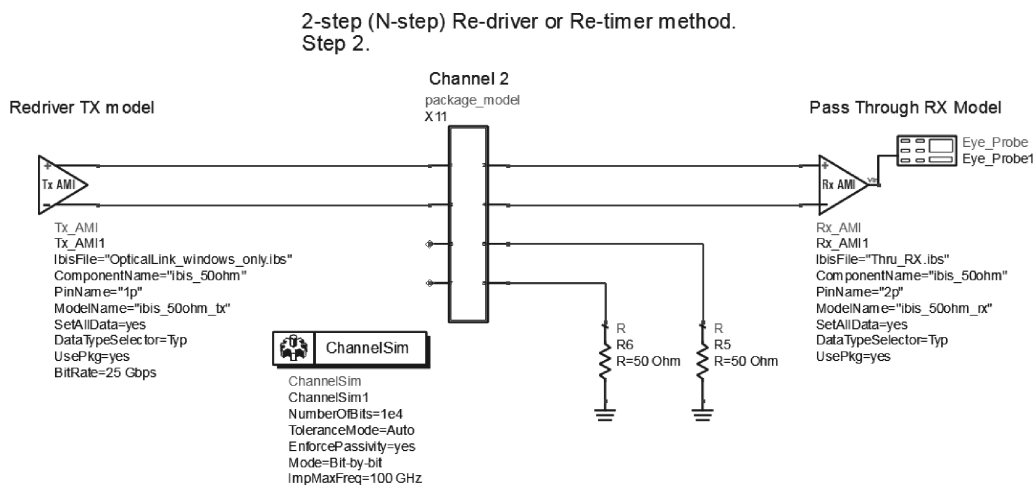


图 9-25 光通道模型作为发射端进行验证

仿真结果如图 9-26 所示。

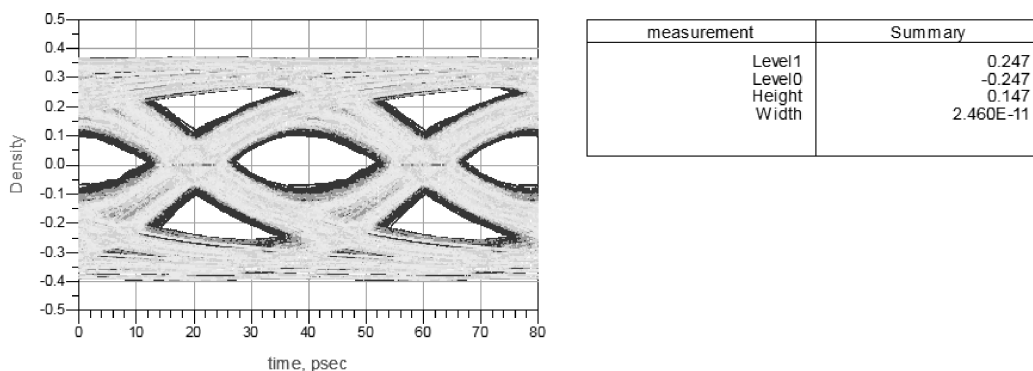


图 9-26 仿真结果

## 2. IBIS - AMI 模型作为重驱动模型进行验证

最后，将光通道的 IBIS - AMI 模型作为重驱动模型，放置在传输通道之中进行验证，如图 9-27 所示。

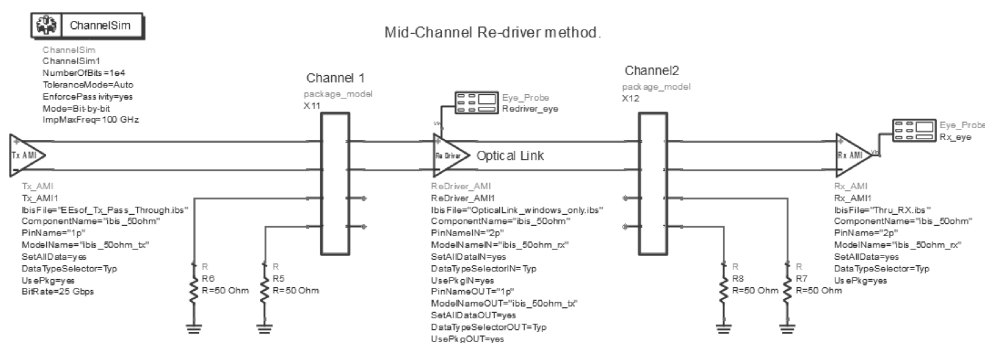


图 9-27 对整个传输通道仿真

图 9-28 所示为加入光通道重驱动模型后的整个通道相应。

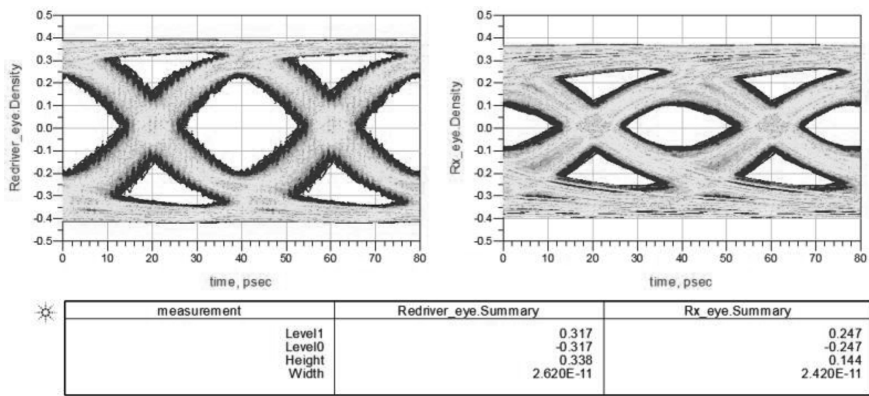


图 9-28 仿真结果

可见，经过光通道以后，信号质量有所下降。

在光通道 IBIS - AMI 模型的 AMI 页面下，可以看到在 SystemVue 中定义的各个器件的对应参数，如图 9-29 所示。

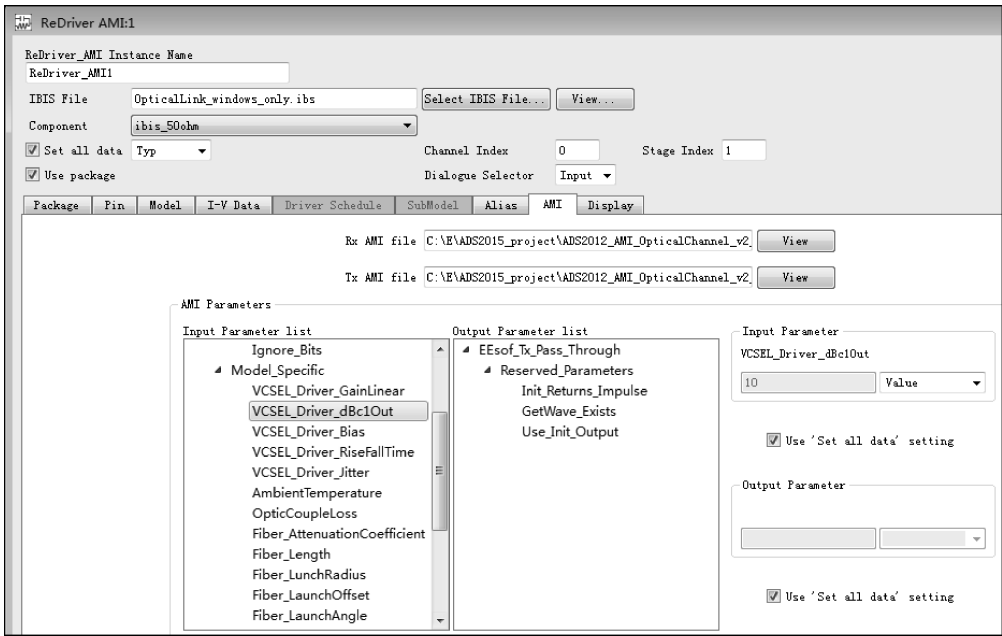


图 9-29 IBIS - AMI 模型中参数设置页面

改变其中的参数，可以观察到仿真结果的变化。

**【总结】**

经过本实例，在 SystemVue 中进行了通用光通道的 IBIS - AMI 模型创建，并在 ADS 中使用通道仿真器进行了验证。

光模块供应商可以参考此设计流程进行其自定义光模块的 IBIS - AMI 模型创建及验证工作。

# 实例 10 Cadence Allegro 版图的提取、导入及仿真

## 【目的】

本实例主要介绍如何提取使用 Cadence Allegro™ 软件绘制的版图的关键走线，并将提取的版图导入到 ADS 软件中进行仿真的过程。

Cadence Allegro™ 是常用的 PCB 布线工具。ADS 软件提供 ADFI (Allegro Design Flow Integration) 功能用来导入 Allegro 软件绘制的版图，其中包含安装在 Allegro 软件中的插件以及 ADS 软件中的导入工具。本实例中将介绍此插件的安装及使用，以及将提取出的版图导入到 ADS 软件中进行仿真的流程。

本实例中使用的是 Cadence Allegro 16.6 版本以及 ADS2014.01 版本，实例中使用的内存条版图 PC3-10600-UDIMM-V0\_50\_RC\_Fx\_20070530.brd，可从 JEDEC 网站上下载：[https://www.jedec.org/sites/default/files/docs/design/DDR3/PC3\\_10600\\_UDIMM\\_V050\\_RC\\_F0\\_20070530.zip](https://www.jedec.org/sites/default/files/docs/design/DDR3/PC3_10600_UDIMM_V050_RC_F0_20070530.zip)。



## 10.1 AllegroDFI 插件的安装及设置

启动 Allegro PCB editor，单击菜单 File→Script...，找到 ADS 安装目录下 ial\scripts 子目录的 eemLocalConfig.scr 文件，按照图 10-1 ~ 图 10-4 所示进行操作。



图 10-1 AllegroDFI 插件安装（一）

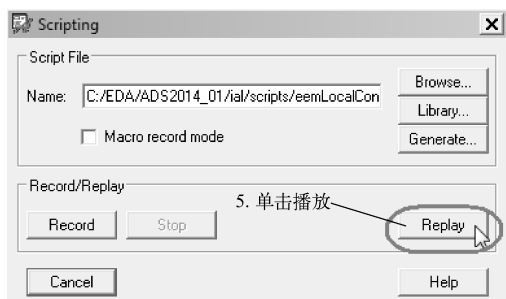


图 10-2 AllegroDFI 插件安装 (二)

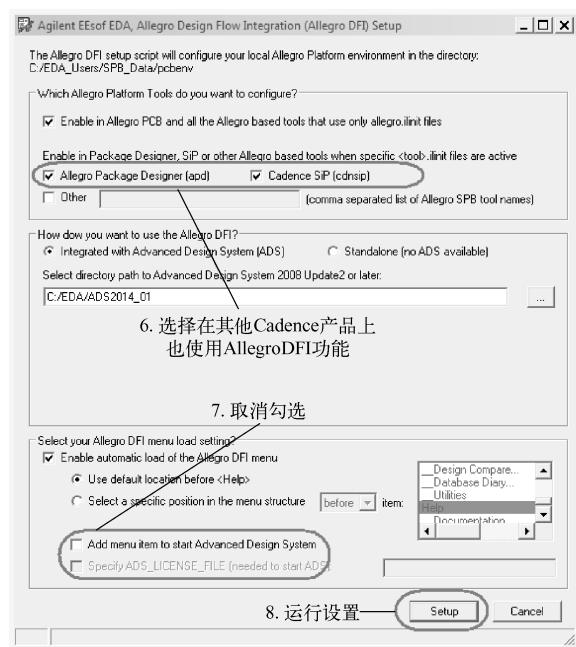


图 10-3 AllegroDFI 插件安装 (三)

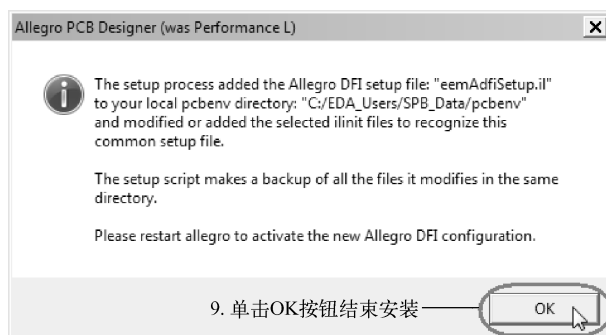


图 10-4 AllegroDFI 插件安装 (四)

安装完成后，退出 Allegro 软件。重新启动 PCB Editor，如果安装正确，会出现如图 10-5 所示的变化。

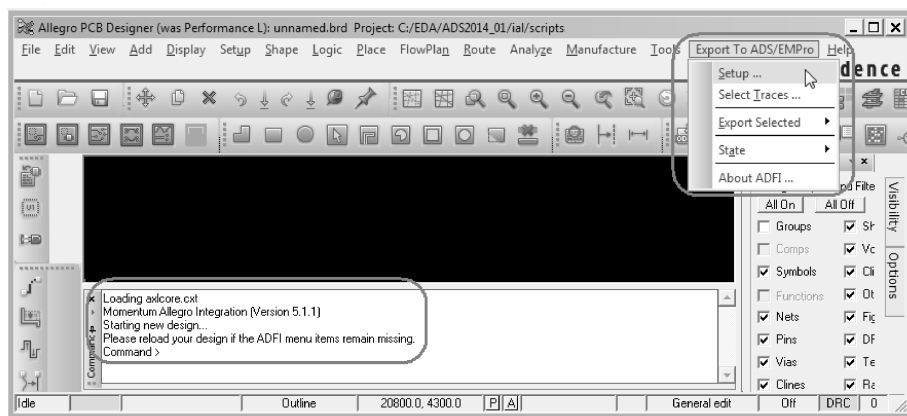


图 10-5 AllegroDFI 正确安装后的界面变化

至此，AllegroDFI 所需的插件设置完毕，可以开始进行版图的提取和导出。



## 10.2 在 Allegro 软件中使用 AllegroDFI 插件进行版图提取

(1) 在 Allegro 软件中打开版图设计。本例中打开版图文件：PC3 - 10600 - UDIMM - V0\_50\_RC\_Fx\_20070530. brd。

(2) 在 Allegro 版图设计窗口查看版图设计单位：如图 10-6 所示，单击 Allegro 版图菜单 Setup→Design Parameters...，选择 Design 栏，本设计中使用设计单位是毫米，在建立 ADS 软件的工程项目时，需要设置相同的设计单位。

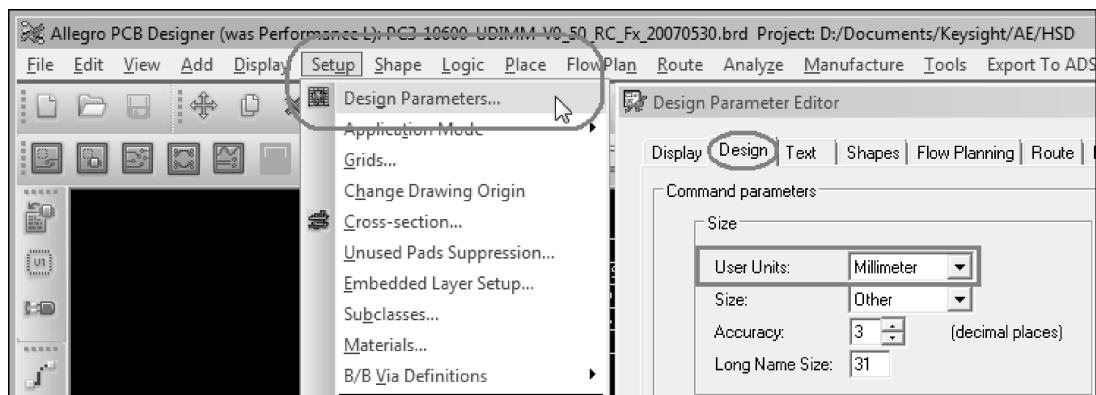


图 10-6 检查 Allegro PCB 版图设计单位设置

(3) 在 Allegro 环境中检查版图截面设置 (Cross Section)：如图 10-7 所示，单击菜单项 Setup→ Cross - section 或者单击快捷图标。

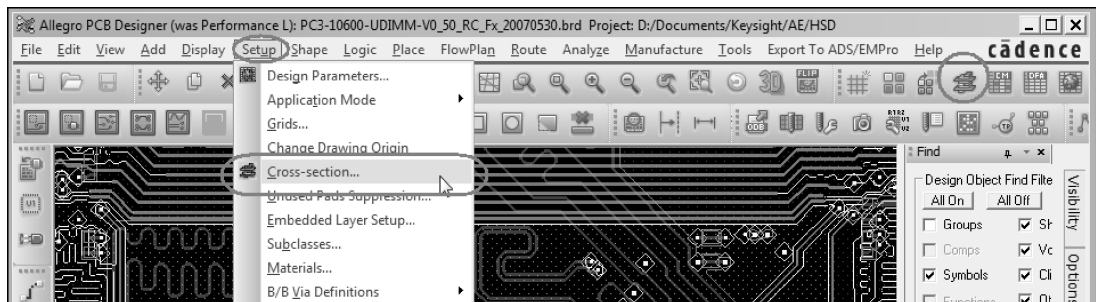


图 10-7 查看 Allegro 版图层叠设置

可以观察到各板层的材料参数，在本实例中没有负片层 (Negative Artwork)，如图 10-8 所示。

(4) 如图 10-9 所示，执行导出命令 Export To ADS/EMPro→Setup...，在输出设置中选择 “Sample Board setting using Strips, Slots and Vias”，并单击 OK 按钮。

在此输出设置中，负片层将被转换为 ADS 软件中的 slot 层。本实例中没有负片层，所有的板层都会被转换为 ADS 软件中的 strip 层，在后面的内容中还会解释 strip 和 slot 层的区别。

	Subclass Name	Type	Material	Thickness (MM)	Conductivity (mho/cm)	Dielectric Constant	Loss Tangent	Negative Artwork	Shield	Width (MM)
1		SURFACE	AIR			1	0			
2		DIELECTRIC	CONFORMAL_COA	0.01	0	3.75	0.0035			
3	TOP	CONDUCTOR	COPPER	0.045	595900	3.75	0.0035	<input type="checkbox"/>		0.100
4		DIELECTRIC	FR-4	0.1	0	4.2	0.035			
5	SPLIT2	PLANE	COPPER	0.035	595900	4.2	0.035	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
6		DIELECTRIC	FR-4	0.125	0	4.2	0.035			
7	S3	CONDUCTOR	COPPER	0.035	595900	4.2	0.035	<input type="checkbox"/>		0.100
8		DIELECTRIC	FR-4	0.59	0	4.2	0.035			
9	S4	CONDUCTOR	COPPER	0.035	595900	4.2	0.035	<input type="checkbox"/>		0.100
10		DIELECTRIC	FR-4	0.125	0	4.2	0.035			
11	SPLIT5	PLANE	COPPER	0.035	595900	4.2	0.035	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
12		DIELECTRIC	FR-4	0.1	0	4.2	0.035			
13	BOTTOM	CONDUCTOR	COPPER	0.045	595900	3.75	0.0035	<input type="checkbox"/>		0.100
14		DIELECTRIC	CONFORMAL_COA	0.01	0	3.75	0.0035			
15		SURFACE	AIR			1	0			

图 10-8 Allegro 版图层叠设置

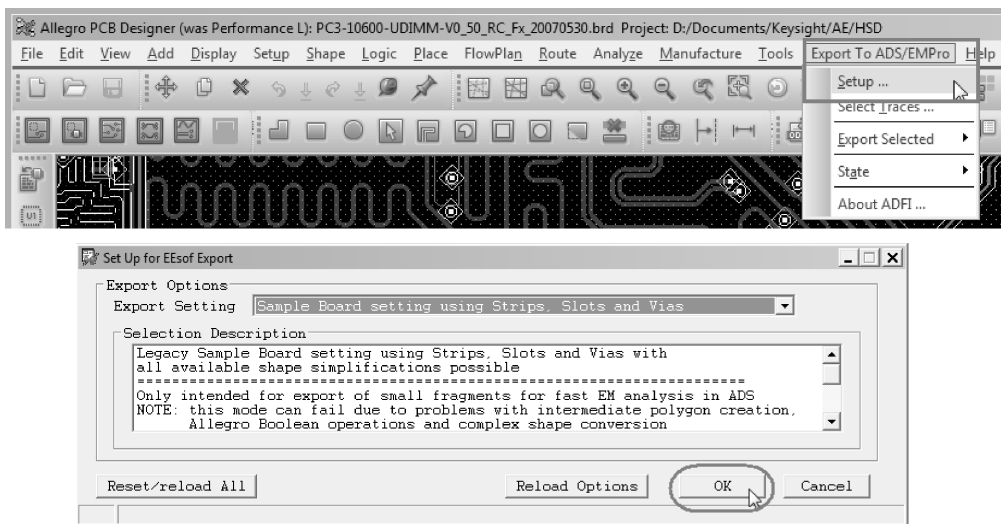


图 10-9 设置 Allegro 版图导出选项

(5) 执行导出命令 Export To ADS/EMPro→Select Trace…。

① 选择走线 (trace) ——选出感兴趣的网络 (nets): 在本实例中, 只选出 DQS0、DQS0#、DQS0R 和 DQS0R#等 4 个网络点。有下面几种方法可以选出网络:

- ☺ 从 Allegro 版图中直接点选网络。这是最常用的方法, 可以直接从版图中选出布线网络。如图 10-10 所示, 单击“Pick Nets”, 在版图中选中地线网络, 然后加入到“RF Ground Nets”(射频地网络)栏目中。选择完成所有的网络后, 单击“End”按钮结束。如果选错了, 单击“Clear”按钮清除, 再重新单击“Start”按钮开始选择网络。使用同样的方法将 VDD 网络加入到“RF Ground Nets”栏目中。
- ☺ 从“Selectable Net Pool”(可选网络池)中选择网络。如果知道希望选择的网络的名字, 可以直接从“Selectable Net Pool”中进行选择。选中后单击“Add”按钮添加到“Signal Nets”(信号网络)或“RF Ground Nets”(射频地网络)中。也可以使用“Filter Nets with Wildcards”(使用通配符过滤)功能, 输入“DQS\*”。按下键盘的

Ctrl 按钮可以进行鼠标多选操作，先按下 Ctrl 按钮，再分别单击 DQS0、DQS0#、DQS0R 和 DQS0R#，并单击中间的“Add”按钮，将选中的走线网络加入到右侧，如图 10-11 所示。

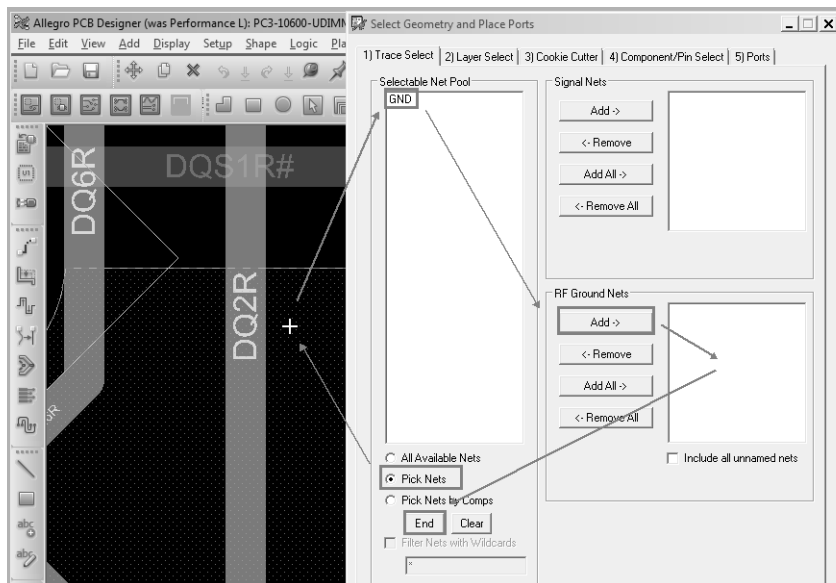


图 10-10 选择地平面

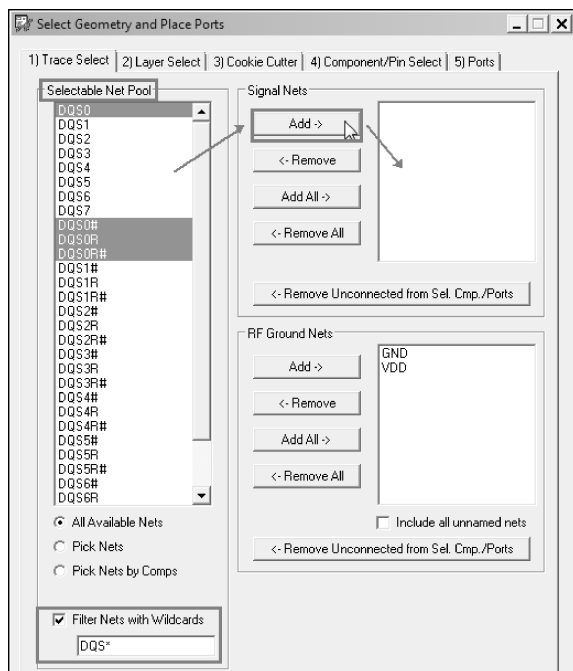


图 10-11 选择信号走线网络

在“RF Ground Nets”中的网络和 ADS 软件电磁场仿真中的“地”不同，在后面会介绍。“Signal Nets”中的网络导出后会在 ADS 软件中加入管脚“Pins”，而“RF Ground Nets”中的网络在导出到 ADS 软件中时不会生成管脚。



② 板层选择：如图 10-12 所示，左边窗口列出了可以导出到 ADS 软件的板层，右边窗口列出将要导出到 ADS 软件中的板层。本实例中选择导出所有板层。

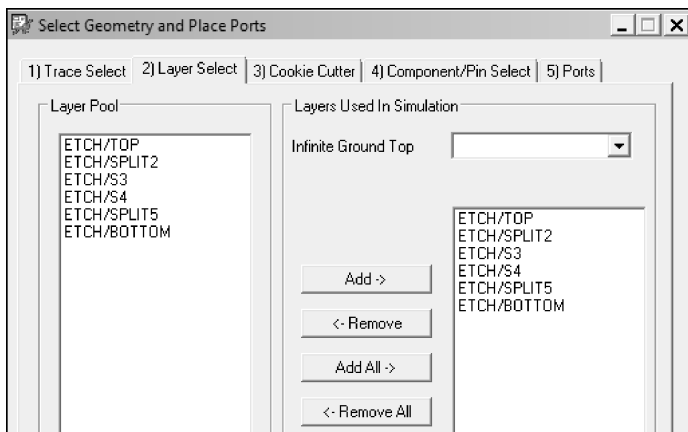


图 10-12 选择输出板层

③ Cookie Cutter (边界裁剪)：如图 10-13 所示。

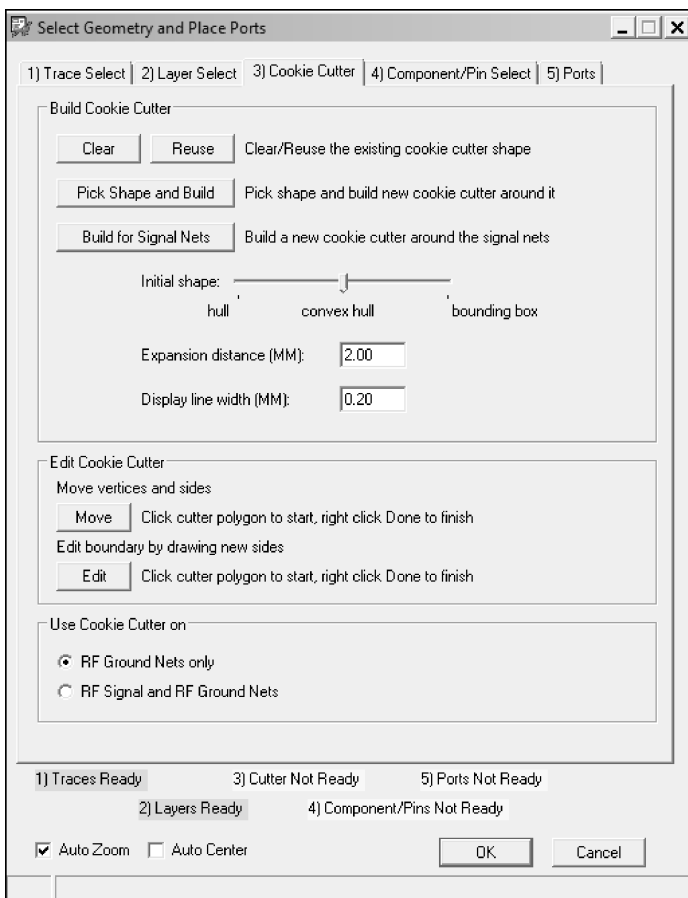


图 10-13 区域裁剪选项设置

- ☺ Expansion distance (扩展距离): 信号网络与边界裁剪区之间的距离。本实例中由 5.0mm 改为 2.0mm。
- ☺ Display line width (显示线宽度): 边界裁剪线的宽度, 本实例中保持默认值 0.20mm 不变。
- ☺ Initial shape (初始形状): 本实例中设定为默认值 “convex hull” (凸形轮廓)。  
当移动鼠标, 将设置由 “convex hull” 向左边 “hull” (轮廓) 方向移动时, 边界裁剪线会紧紧沿着信号网络围住起来。  
当移动鼠标, 将设置由 “convex hull” 向右边 “bounding box” 方向移动时, 边界裁剪线会以直线形式围住信号网络。
- ☺ Use Cookie Cutter on RF Ground Nets Only (只在射频地网络使用边界裁剪): 这意味着边界裁剪线将保留之内所有的信号网络和射频地网络, 并准备输出, 而所有边界裁剪区外边的射频地网络都会被裁剪掉。
- ☺ 单击 “Build for signal nets” 按钮, 就会建立相关的边界裁剪, 如图 10-14 (a) 所示。
- ☺ 当移动鼠标, 将设置由 “convex hull” 向右边 “bounding box” 方向移动时, 再次按下 “Build for signal net” 按钮, 会得到矩形边界裁剪形状, 如图 10-14 (b) 所示。

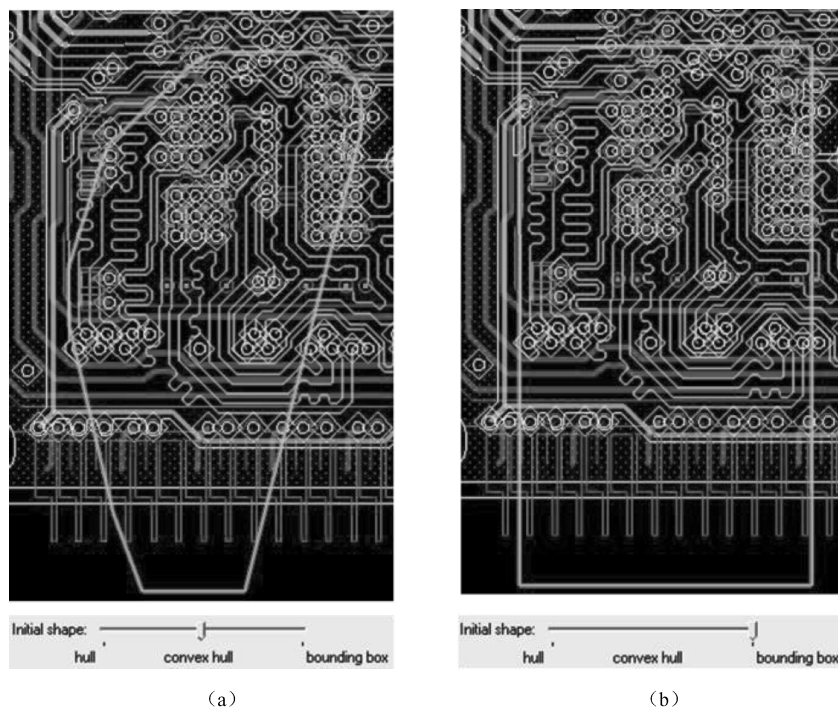


图 10-14 对输出区域进行裁剪调整

- ☺ 如图 10-15 所示, 编辑边界裁剪区:
  - ◇ “Move” 按钮按下后, 先用鼠标单击边界裁剪线, 然后就可以拖拉裁剪线的边或者是线段顶点。单击鼠标右键可以选择结束 (done)、取消 (cancel) 等操作。
  - ◇ “Edit” 按钮按下后, 先用鼠标单击边界裁剪线, 然后再将鼠标移动到希望编辑的裁剪线上, 单击鼠标左键开始编辑线段走向, 单击鼠标右键可以选择结束

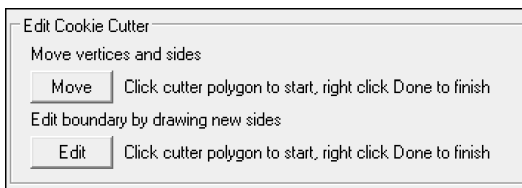


图 10-15 裁剪工具选项设置

(done)、取消 (cancel) 等操作。

◇ 如果希望放弃所做的移动或修改工作，再次单击“Build for signal nets”按钮，可以重新绘制边界裁剪线。

④ (可选操作) 输出边界裁剪区内的所有网络：如图 10-16 所示，单击“Add All”按钮，将左边窗口中所有的网络加入到“RF Ground Nets”窗口中。所有的信号网络 (Signal Nets) 都会自动的加上管脚 (Pins)，所有的射频地网络不会加上管脚。这样，边界裁剪区内的所有网络都会被输出，之外的网络将会被裁剪掉，不会被输出。

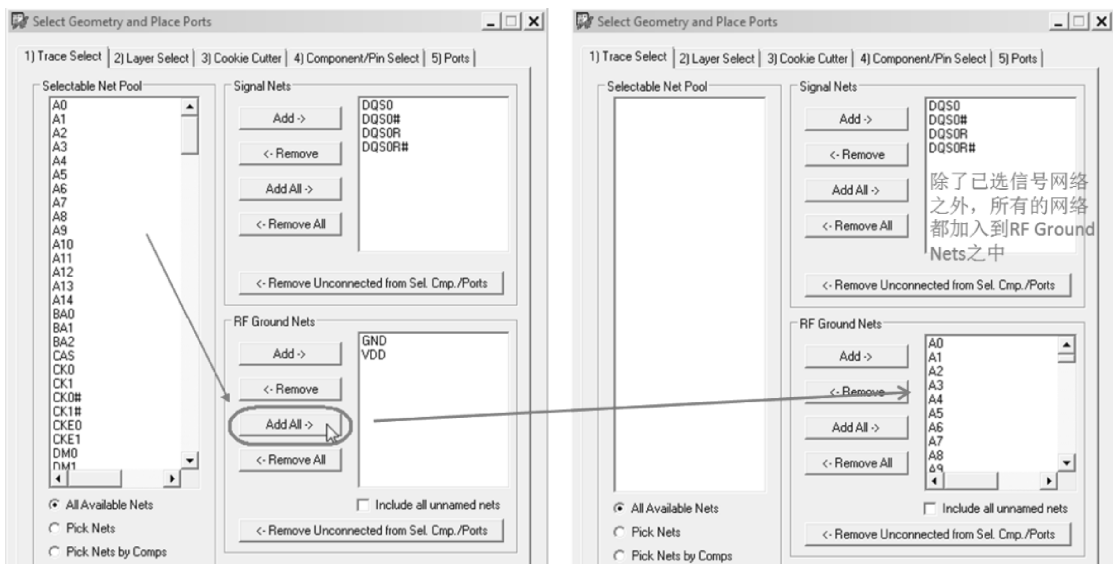


图 10-16 版图输出网表设置

⑤ 元件及管脚 (Component/Pin) 选择：所选网络的管脚名字会自动地填入右下角的窗口中。如果不希望将所有的管脚输出到 ADS 软件中，可以选中不希望输出的管脚，将它们移除到右下角的窗口中。在本实例中，不需要进行元件的输出，将元件列表中的元件移除掉。另外，所有的管脚应该会自动包含进来，请与图 10-17 进行比较确认。

⑥ 端口 (Port)：确认勾选“Do not add negative ref. pins” (不加入负参考管脚) 选项；单击“AutoPlace”按钮；单击“+”，可以观察到 Negative Reference Pins 显示“No ref. Pin defined” (未定义参考管脚)；单击 OK 按钮，如图 10-18 所示。

(6) 执行 Allegro 菜单命令 (Export To ADS/EMPro→Export)：单击 Allegro 菜单 Export To ADS/EMPro→Export Selected→To ADS...，保存 \*. adfi 文件，该文件包括了版图层叠、端口、材料和所有的版图信息，如图 10-19 所示。

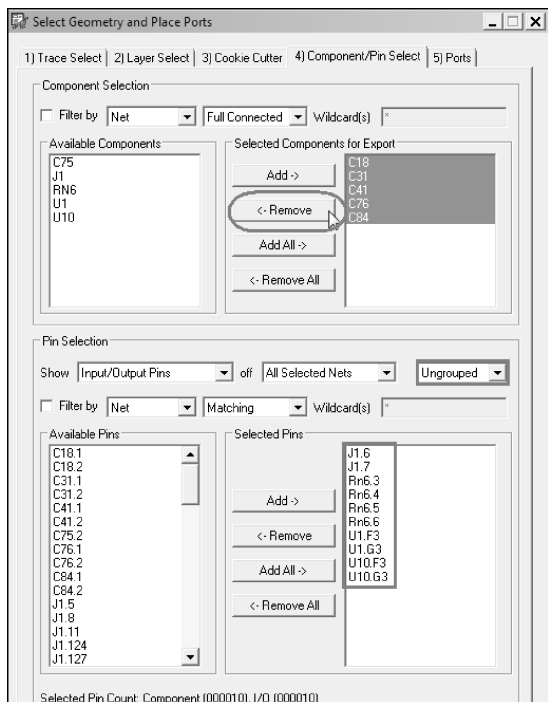


图 10-17 版图输出元件及管脚设置

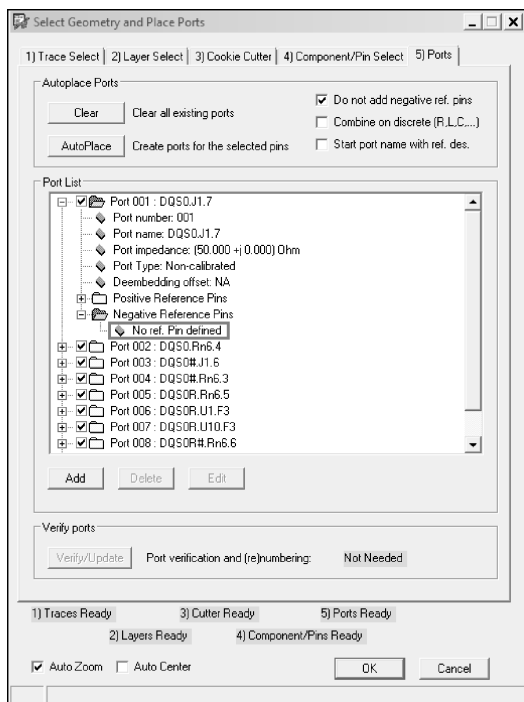


图 10-18 端口设置

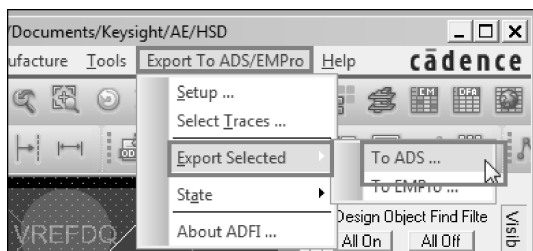


图 10-19 导出所选区域

在计算机本地硬盘中会保存导出的文件，如图 10-20 所示。

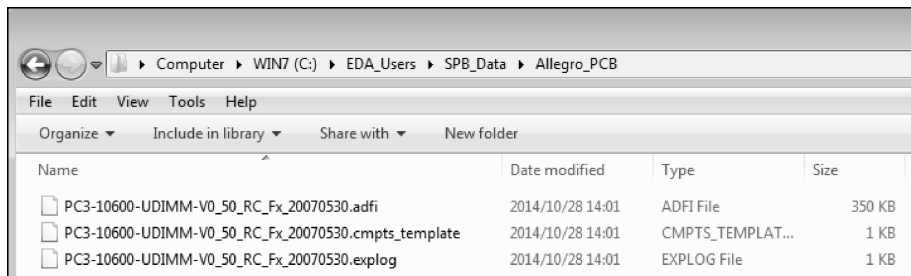


图 10-20 版图输出文件信息

(7) 执行 Allegro 菜单命令 (Export To ADS/EMPro→State):

- ☺ 使用保存状态功能，可以保存所有在菜单项 “Select Traces” 中进行的设置，如图 10-21 所示。
- ☺ 使用调用状态功能，恢复保存的 “Select Traces” 中的设置参数，如图 10-22 所示。

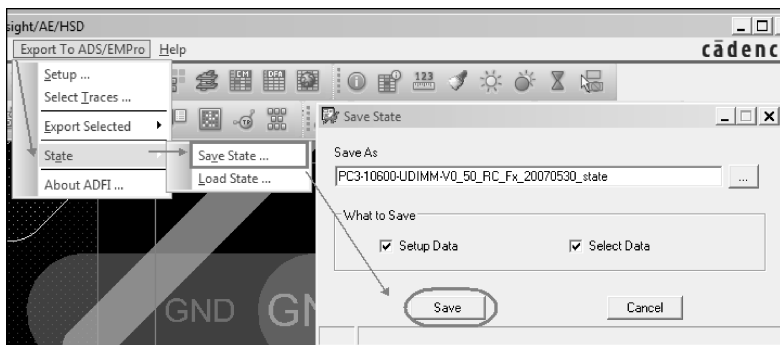


图 10-21 保存版图输出设置

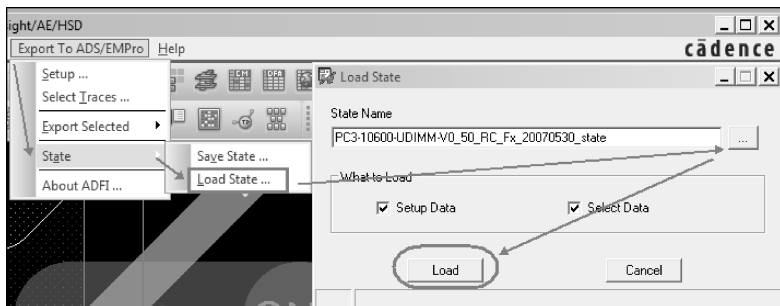


图 10-22 调用版图输出设置



### 10.3 创建 ADS 工程，导入 Allegro 版图

(1) 激活 ADFI 附加工具：启动 ADS 软件，单击主窗口中的 Tools→Manage AEL Addons..., 如图 10-23 所示。

勾选 ADFI Import Tools，并确认激活该功能，如图 10-24 所示。

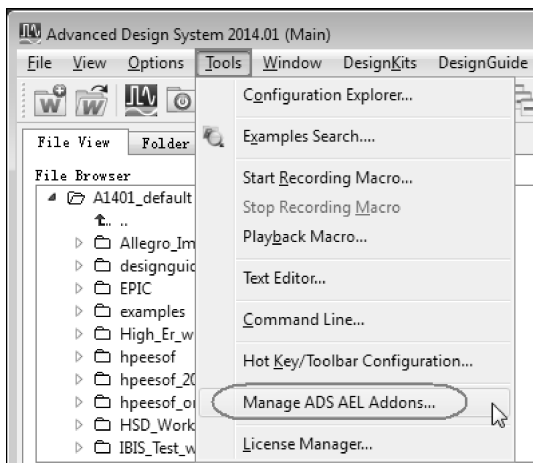


图 10-23 打开 AEL 工具管理菜单

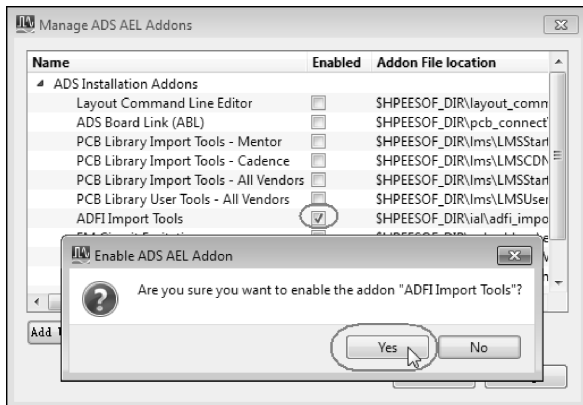


图 10-24 激活 ADFI 导入工具

信息提示,在下次 ADS 软件启动后,将会启用该功能,如图 10-25 所示

(2) 重新启动 ADS 软件,创建名为 Allegro\_Import 的工程,注意选择与 Allegro 版图设计中相同的版图单位,本实例中使用的单位是 millimeter。

(3) 在 ADS 软件主窗口中,如图 10-26 所示,单击菜单 File→Import→Design...。

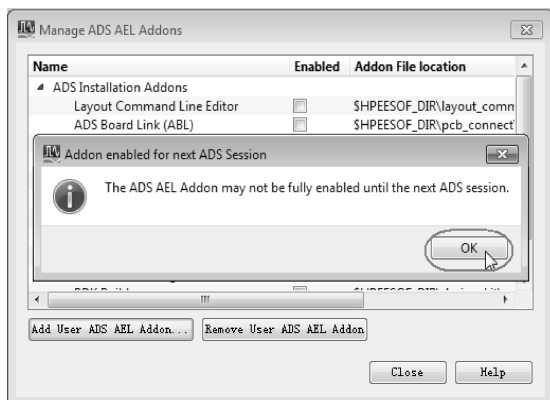


图 10-25 新的设置需要重新启动软件才会生效

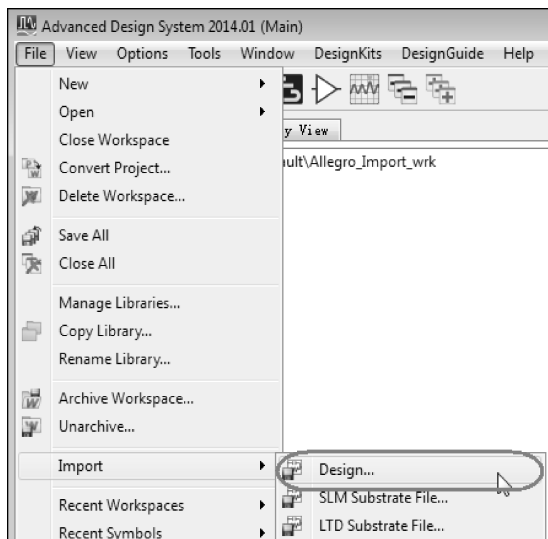


图 10-26 ADS 软件的版图导入选项

在 FileType 中选择最后一项 ADFI File Format,并在 Import file name 选项中指定输出的版图文件的位置,单击 OK 按钮确认导入,如图 10-27 所示。

单击 Options,本实例中使用如图 10-28 所示的选项。

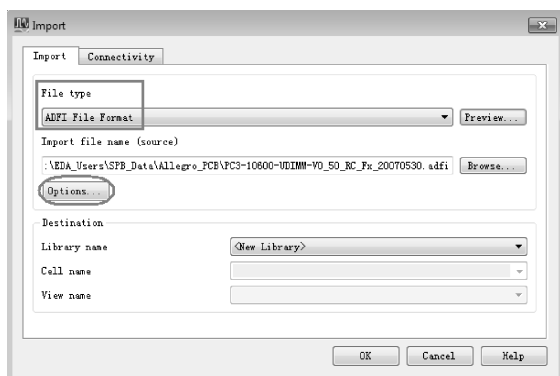


图 10-27 ADFI 版图导入窗口

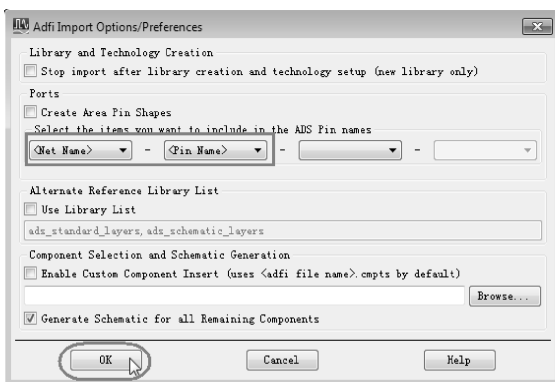


图 10-28 ADFI 版图导入选项

设置完成后,单击 OK 按钮回到导入界面,并再次单击 OK 按钮开始导入版图。

导入到 ADS 软件的版图及 ADS 软件工程目录结构如图 10-29 所示,由于之前激活了 ADFI 附加功能,所以在版图窗口菜单栏中可以看到 ADFI Tools 菜单项。

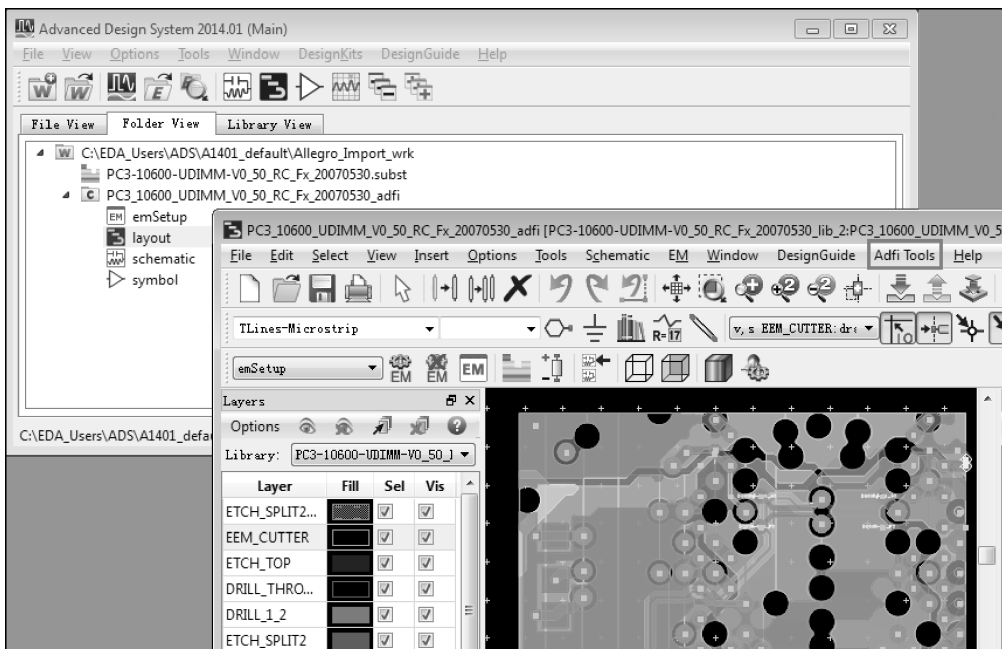


图 10-29 导入的 ADS 软件版图及工程目录结构



## 10.4 查看导入的版图

(1) 自定义快捷键：使用快捷键可以方便操作，如使用 L 键替换在本实例中经常用到的菜单功能 Options→Layer Preferences…。图 10-30 给出了快捷键的定义过程，在版图窗口中单击菜单项 Tools→Hot Key/Toolbar Configuration…进行快捷键定义窗口，图 10-30 将功能项目赋予按钮“L”。

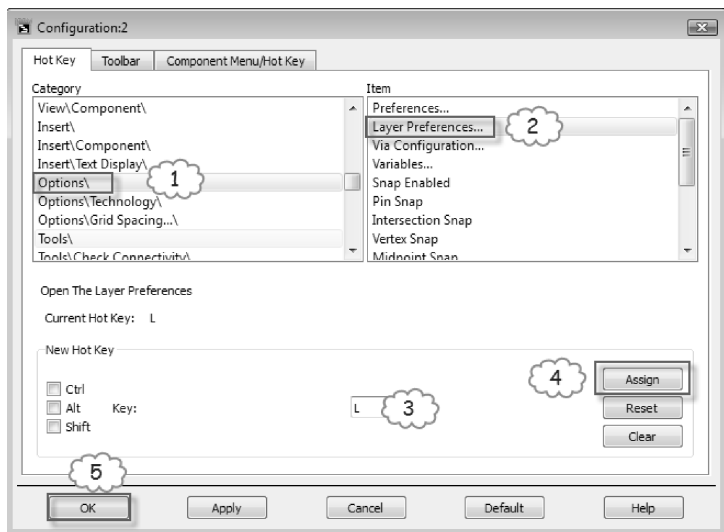


图 10-30 快捷键设置

快捷键设定完成后,单击菜单项 Options 就可以看到 Layer Preferences...项目旁边的快捷键指示标志,并且当按下“L”按钮后,就会弹出板材层设置的界面,如图 10-31 所示。

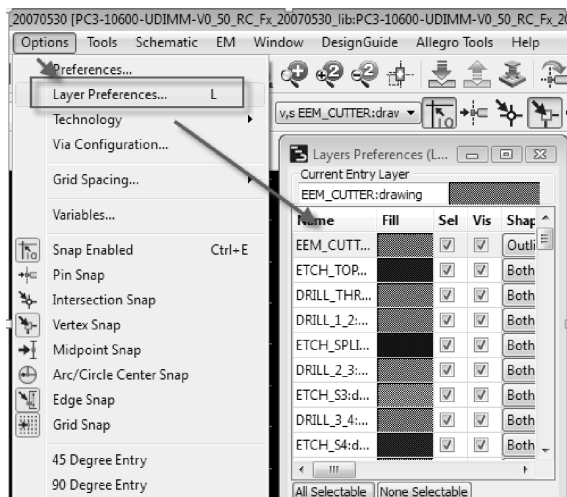


图 10-31 板层设置

使用同样的方法设置其他 4 个快捷键 (在 View/LayerView 选项中), 如图 10-32 所示。

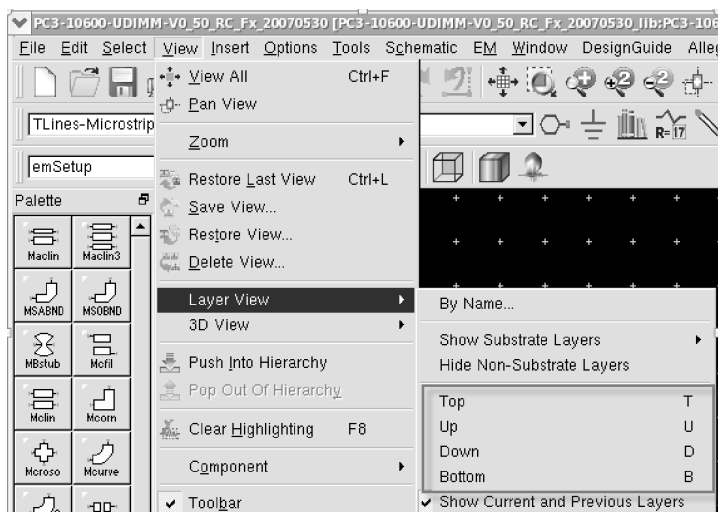


图 10-32 板层切换快捷键

设置完成后,可以对这些按钮进行测试,快捷键没有大小写之分:

- ☺ 按下“L”键,观察是否弹出“Layers Preference”窗口。
- ☺ 按下“T”键,可以看到版图窗口中只显示最上层版图。
- ☺ 按下“D”键,可以看到第一个过孔层或者介质层。
- ☺ 再次按下“D”键,可以看到第二层金属层,继续按下“D”键,直到最后一层。
- ☺ 再试试“B”键和“U”键。

(2) 版图三维显示: 打开 PC3\_10600\_UDIMM\_V0\_50\_RC\_Fx\_20070530\_adfi 单元下的 Layout 视图,单击菜单栏上 EM 设置工具条,如图 10-33 所示。





图 10-33 电磁场仿真设置快捷按钮

在 EM 设置菜单中，确认仿真引擎为 MomentumRF，如图 10-34 所示。

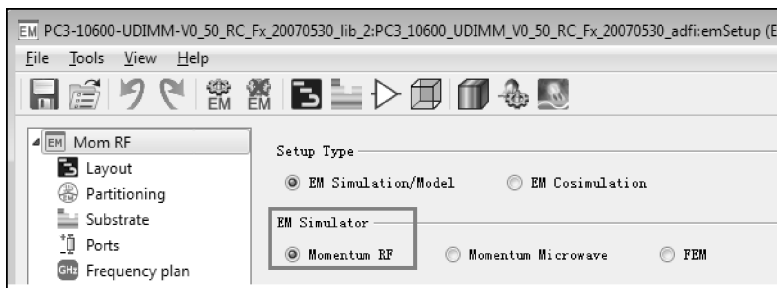



图 10-34 电磁场仿真器设置

如图 10-35 所示，单击 EM 设置界面中的 Layout 项，并单击页面中的“Show 3D View”按钮，或者单击工具条中的三维预览按钮。之后就可以看到三维预览图，如图 10-36 所示。

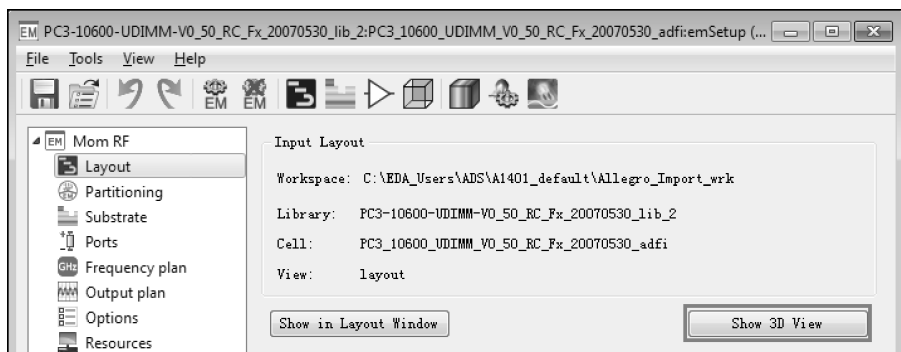


图 10-35 三维预览按钮

在三维预览窗口可以进行下面的操作：

- ⑤ 尝试使用各个工具图标：当鼠标移动并停留在某个工具图标上时，会显示此工具图标的功能，如图 10-37 所示。
- ⑤ 在 ADS 软件的版图窗口、原理图窗口、数据显示窗口以及三维预览窗口都可以使用鼠标按钮功能。
  - ◇ 使用鼠标中间的滚轮可以对三维视图进行放大缩小。
  - ◇ 按下鼠标右键，并移动鼠标，可以摇动观察窗口。
  - ◇ 按下鼠标左键，并移动鼠标，可以旋转观察的物体。

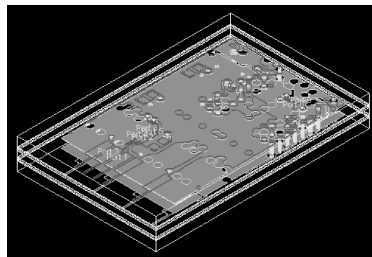


图 10-36 三维预览图

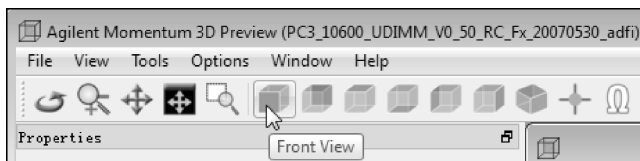


图 10-37 工具条及功能说明

- ⑤ 使用菜单项 Tools→Z Scale..., 并将滑动条从 1 拉向 10, 可以将视图的垂直方向进行拉伸以观察细节, 如图 10-38 所示。

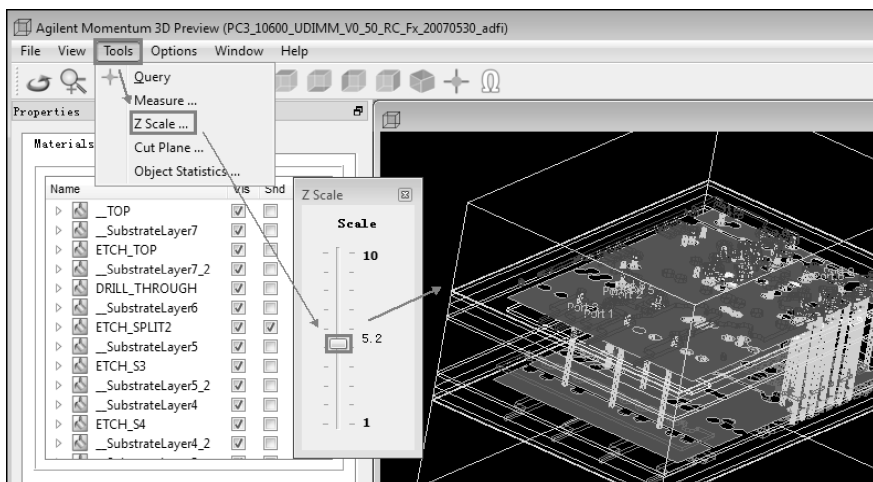


图 10-38 垂直方向拉伸滑动条

- ⑥ 按下鼠标左键旋转整个电路板, 观察细节。  
 ⑦ 在左侧的板层设置中选中 ETCH\_TOP, 单击“Color”按钮, 可以选择不同的颜色。  
 ⑧ 在 Transparency 设置中, 将滑动条从 0% 设置到 50% 左右, 板材 ETCH\_TOP 就会变成半透明, 可以观察到其中的过孔, 如图 10-39 所示。

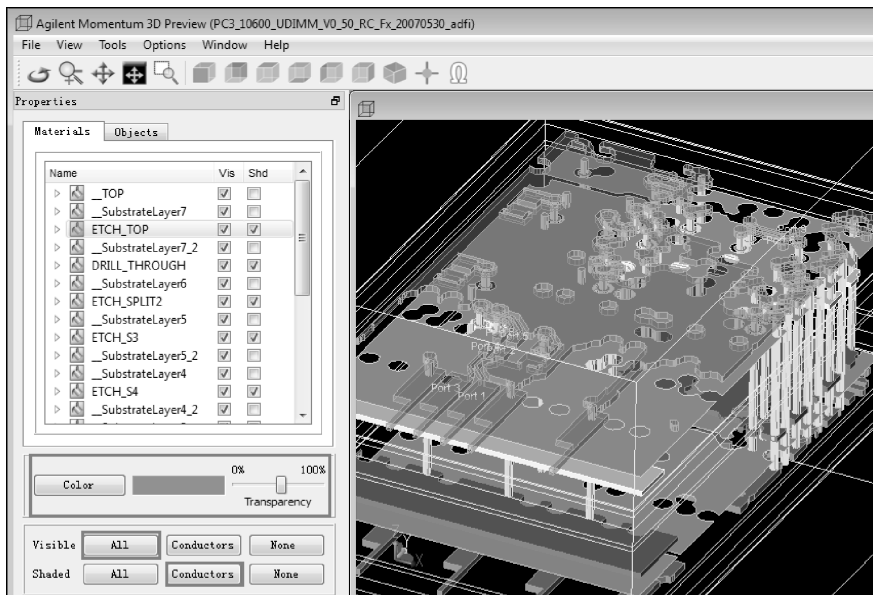



图 10-39 透明度设置

(3) 版图连接性检查：在版图窗口中可以进行二维连通性检查。在版图窗口中，按下之前定义好的快捷键“L”，再单击“Make All Selectable”按钮，如图 10-40 所示。

之后单击版图窗口菜单项 Tools→Check Connectivity→Show Nodal Interconnect，如图 10-41 所示。

使用鼠标单击如图 10-42 所示的过孔处，就可以看到整条与此连接的走线。

使用版图窗口侧边的 Navigator 功能，选择走线网络名称，也可以在版图中对该走线网络进行高亮显示。Navigator 窗口中的设置按钮，可以进行操作选项设置，如图 10-43 所示。

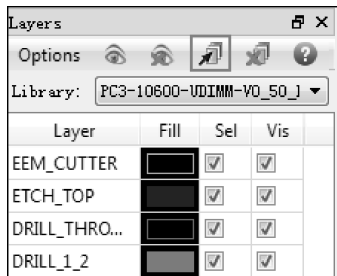


图 10-40 板层显示/可选性快捷按钮

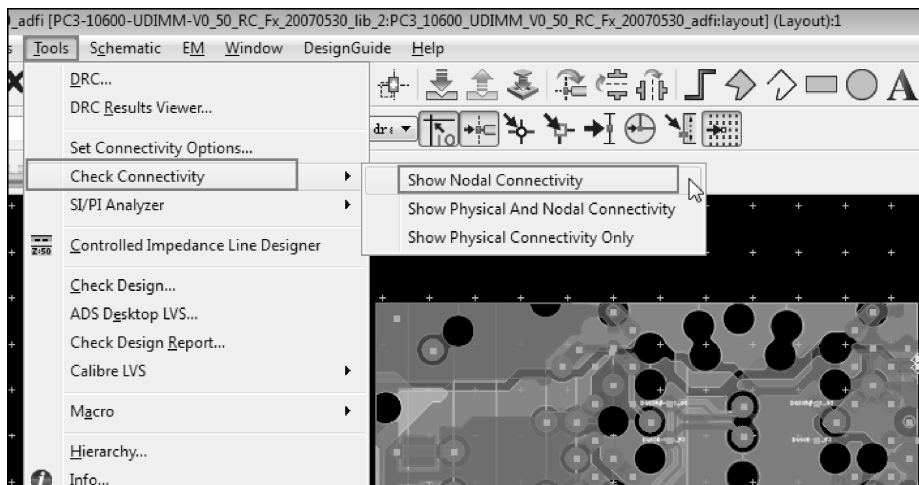


图 10-41 连通性检查

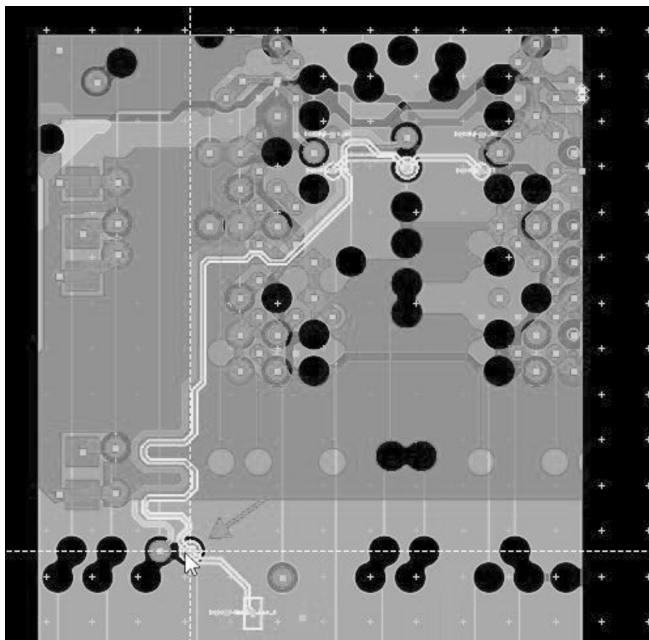


图 10-42 高亮显示连通的布线

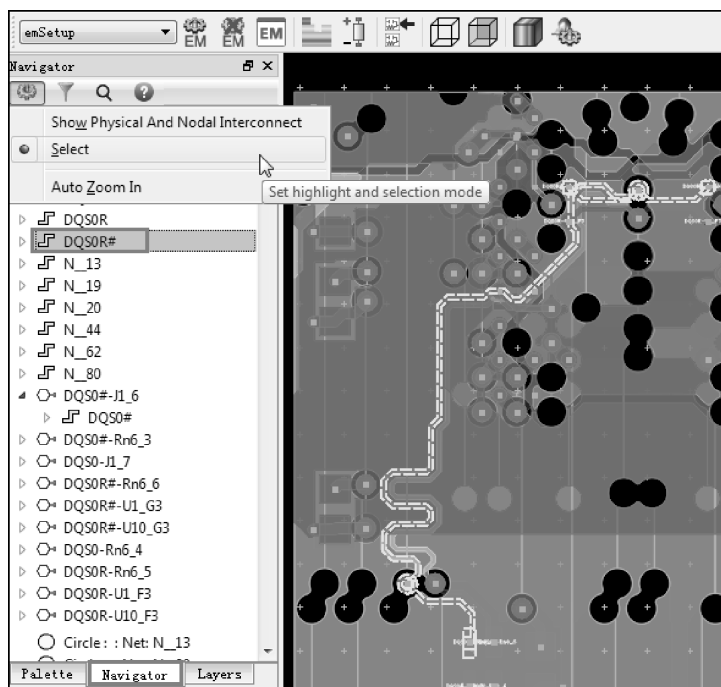


图 10-43 高亮显示选中节点网络



## 10.5 Momentum 设置及仿真

### 1. Momentum 介质/板层设置

在 EM 设置窗口中单击“Substrate”就可以看到导入的介质/板层设置，单击窗口中的“Open...”按钮，打开层叠设置，如图 10-44 所示。

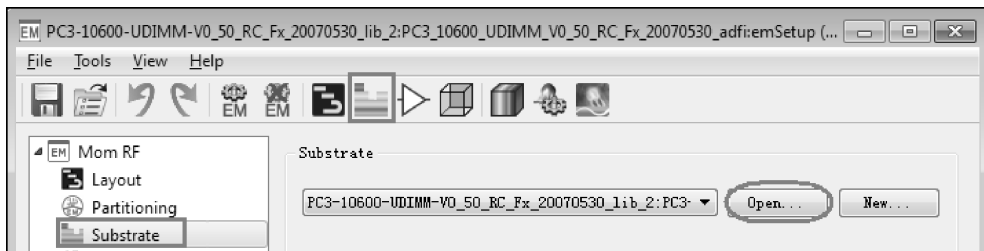


图 10-44 打开层叠设置界面

在打开的层叠设置界面中可以看到所有的板层以及 5 个盲孔层和一个过孔层，如图 10-45 所示。

单击任意一个过孔层观察过孔层属性，如图 10-46 所示。

金属走线的厚度是 0.045mm，一般情况下，如果金属厚度/走线宽度  $> 0.2$ ，这样的金属可以称之为厚金属。对于厚金属，可以使用“Expand the substrate”选项，并选择在介质

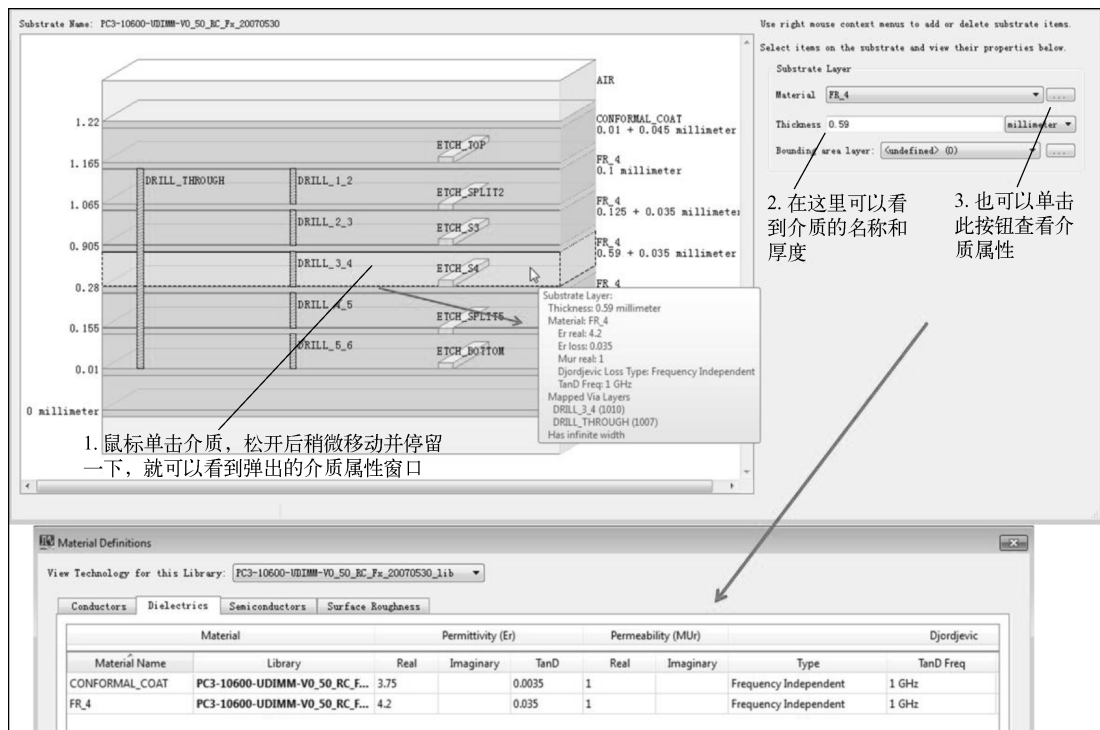


图 10-45 层叠设置及材料属性选择

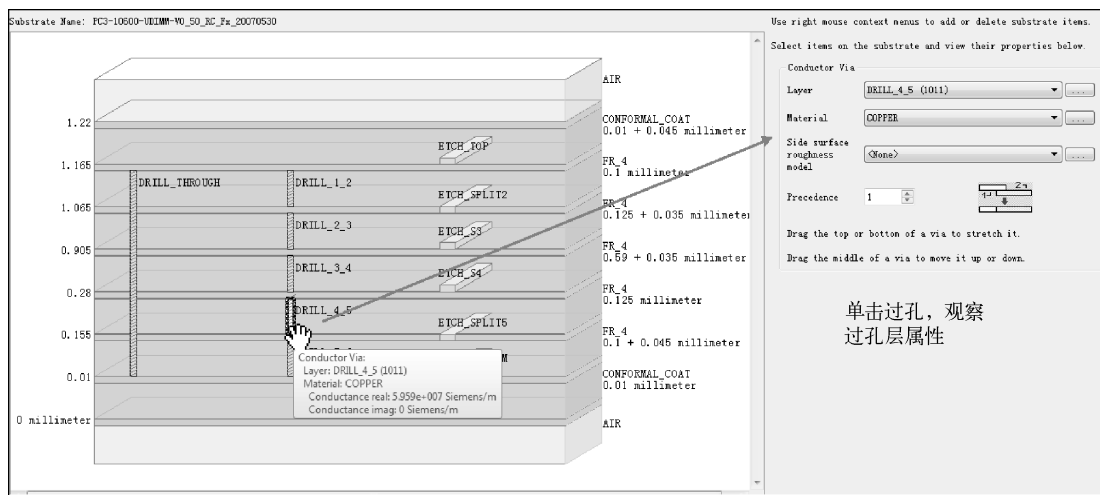


图 10-46 过孔设置

之上还是介质之下，如图 10-47 所示。使用这两种设置，上层或者下层的介质将会被“垫”高。

也可以使用“Intrude into substrate”选项，并选择 above interface 或者是 below interface。使用这两种设置，介质的高度不会发生变化，如图 10-48 所示。

现在回到版图窗口，测量一下走线的宽度：

首先单击快捷键“L”调出 Layers 窗口，单击快捷按钮 关闭所有层，之后只打开 ETCH\_S3 层，如图 10-49 所示。

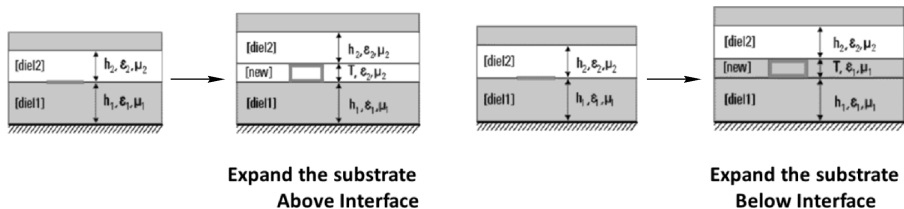


图 10-47 厚金属层设置

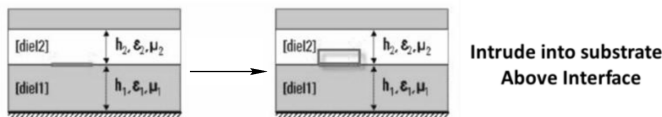


图 10-48 金属层“嵌入”到介质中

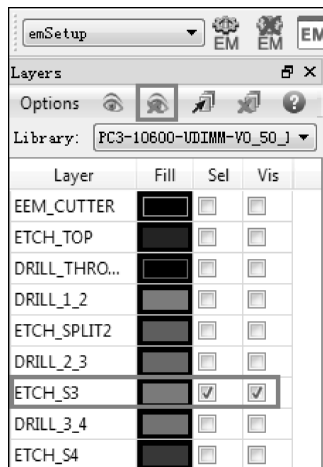


图 10-49 关闭所有板层显示快捷按钮

在版图窗口快捷工具条上设置版图捕获 (snap) 方式, 如图 10-50 所示, 加框的图标为需要设置的选项。

之后在版图窗口中, 单击鼠标右键, 在右键菜单中可以找到 Measure 命令, 如图 10-51 所示。



图 10-50 版图捕获方式设置快捷按钮

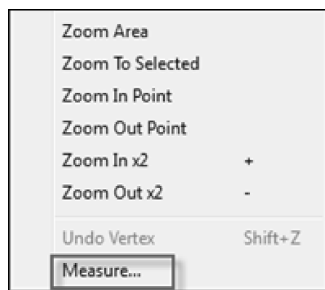



图 10-51 测量菜单

选择 Measure... (或者单击版图窗口菜单 Insert→Measure...), 对 ETCH\_S3 层上的走线进行测量, 如图 10-52 所示。

测量线宽的结果为 0.1mm, 代入公式:  $0.045\text{mm}/0.1\text{mm} = 0.45 > 0.2$ , 因此金属走线应视为厚金属。

将每一层金属都按照 “Intrude substrate” 方式设定, 除了 Etch\_Bottom 之外, 所有的金属层都选择 above interface 方式。

设定完成后, 在单击工具条  保存设置时, 窗口左下方出现出错信息, 如图 10-53 所示。

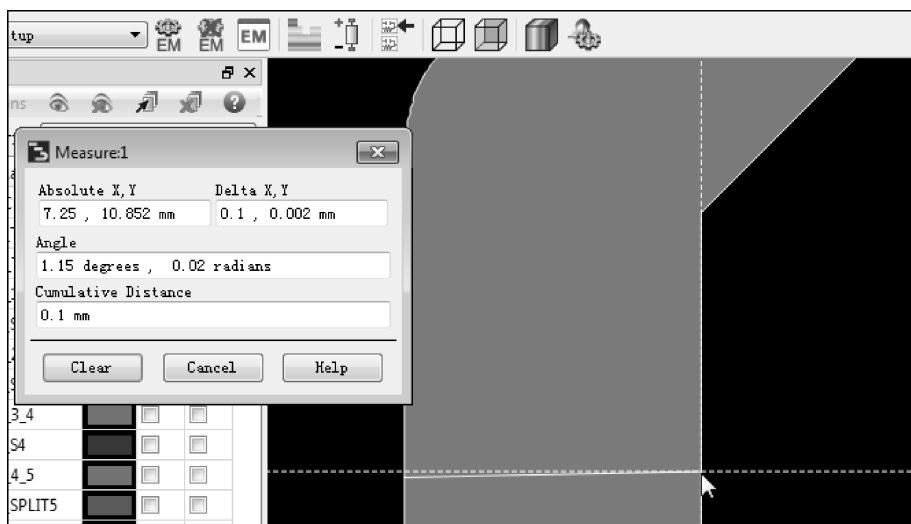


图 10-52 测量走线宽度

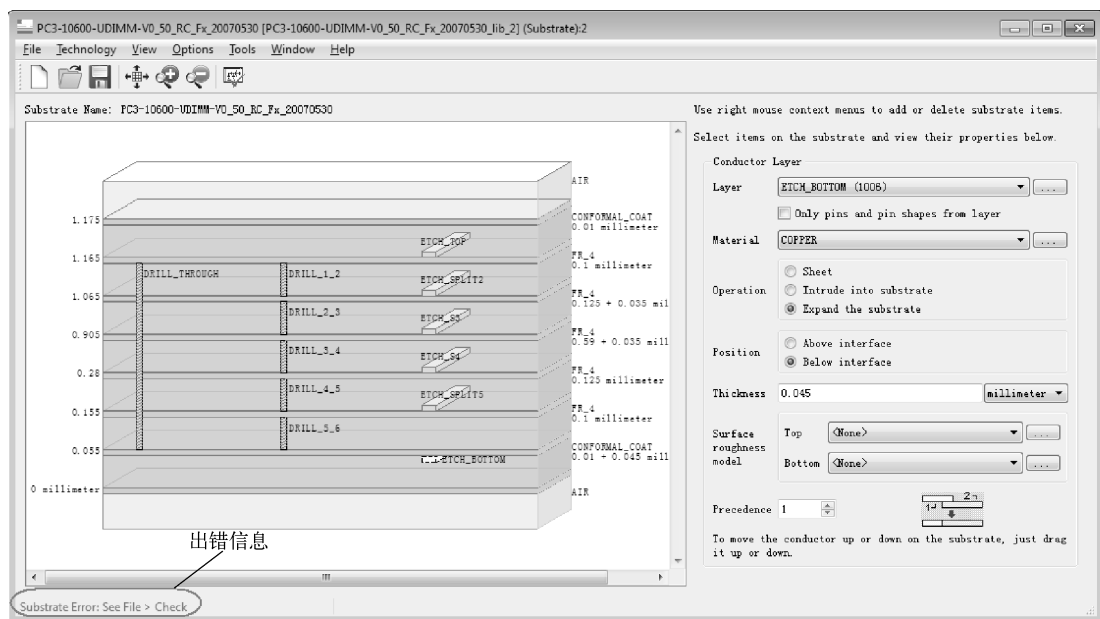


图 10-53 层叠设置及出错信息

依据提示 File→Check, 可以看到下面的信息, 指出顶层金属的厚度大于 conformal coat layer (共形覆盖) 的厚度, 如图 10-54 所示。

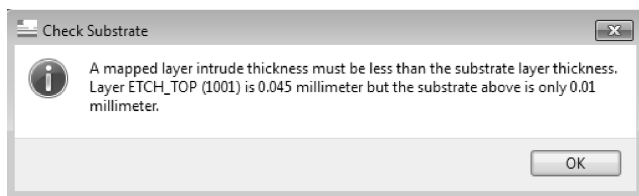


图 10-54 层叠错误信息

Top 层和 Bottom 层只有很少的短线，为了兼顾 Conformal Coat 层，将 ETCH\_TOP 和 ETCH\_BOTTOM 改为薄金属。ETCH\_SPLIT2、ETCH\_S4 和 ETCH\_SPLIT5 没有走线，可以更改为 Sheet（薄片）金属。ETCH\_S3 为唯一的厚金属层。

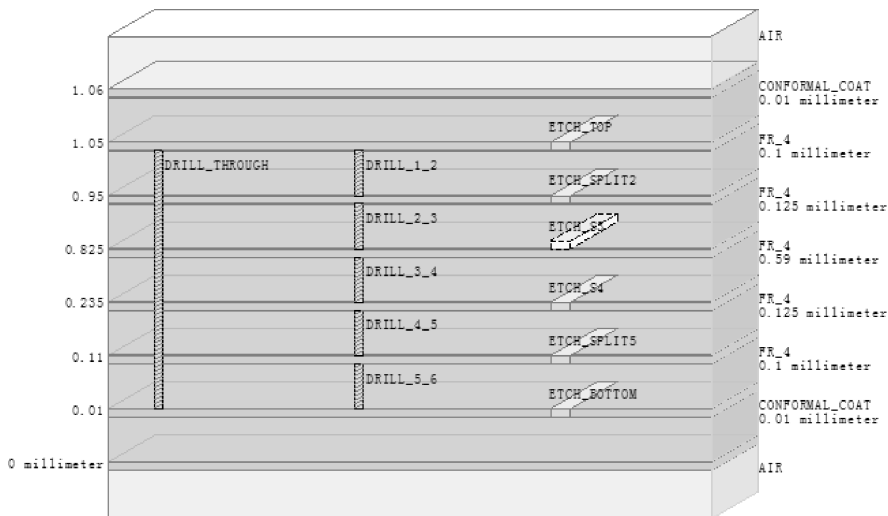



图 10-55 修改后的层叠设置

如图 10-55 所示的设置完成后，错误信息消失。单击  图标，保存板层设置信息。

## 2. Momentum 地平面的设置

在 Momentum 仿真中，有三种方法进行地平面的设置：

- ☺ 地平面在无限远处：在这种情况下，在层叠设置中不需要定义地平面。一般需要对每一个信号端口设定一个与此对应的负参考端口。
- ☺ 无限大地平面：如图 10-56 所示，在这种情况下，仍然可以为每一个信号端口设定对应的负参考端口。

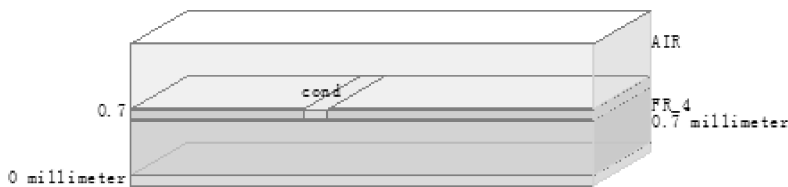


图 10-56 无限大地平面

- ☺ Slot（槽线）层：槽线层可以理解为在无限大地上带有开孔。在这种情况下，也可以在有限地平面上为每个信号端口设定对应的负参考端口。

在本实例中，将使用 Slot 层，不需要设置负参考端口。

通过简单地使用，可以了解 Strip 层和 Slot 层的区别：

在 ADS 主窗口中，新建一个版图，并分别在 Cond 层和 Hole 层中画两个矩形，如图 10-57 所示。同时在 EM 设置中设置板层属性，Cond 层设置为 Strip 层，Hole 层设置为 Slot 层。

使用三维预览功能观察版图，可以看到，设为 Strip 层的板层就是所画的矩形金属片。



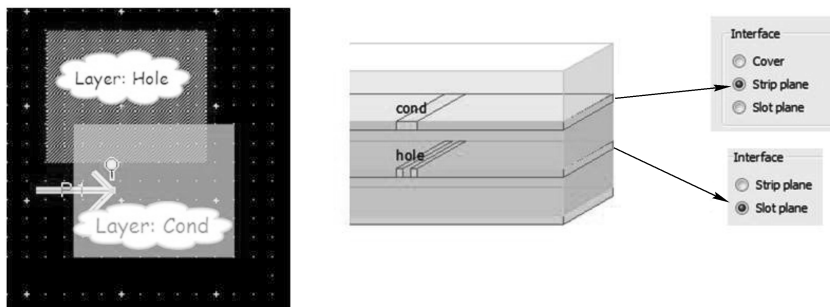


图 10-57 Strip 层与 Slot 层设置

而设为 Slot 层的板层，所画的矩形是一个空洞，而其余的部分是金属，连接到无限大地平面上。如图 10-58 所示，可以看出 Slot 层其实是所画形状的反转图案。

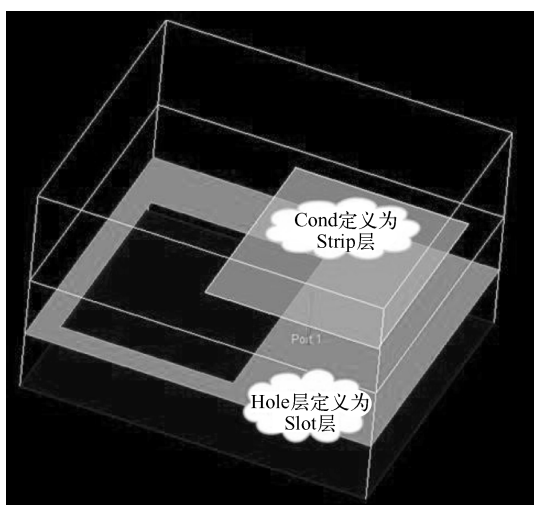


图 10-58 三维显示

回到由 Allegro 导入的版图设计中，在板层设计中，可以选择距离所关心的走线最近的地平面层，将它转为 Slot 层。使用 Slot 层设置，Momentum 只对其中的版图的反片部分进行网格剖分，这样就可以加快仿真速度。

在下面的实验中将会把带有反焊盘的 Strip 层转换为 Slot 层。

我们关心的走线在 ETCH\_S3 层，ETCH\_SPLIT2 层是距离最近的板层，我们将它由 Strip 层转换为 Slot 层。

ETCH\_SPLIT2 层并不是简单的带有反焊盘的地平面，其中包括 GND 和 VDD 网络。如果简单地转换为 Slot 层，GND 和 VDD 网络都会被连接到无限地上。

通常而言，大多数的电路板地层平面上只是一些反焊盘，电源平面一般会在另外的层上。因此我们先假设 ETCH\_SPLIT2 层是带有反焊盘的地平面，看看如何将带有反焊盘的地平面转换为 Slot 层。为了方便起见，我们创建一个新的板层 ETCH\_SPLIT2\_Negative。

(1) 如图 10-59 所示，使用 ADS 版图窗口中的菜单项 Options→Technology→Layer Definitions，在弹出的窗口中单击“Add Layer”创建新的板层并命名为 ETCH\_SPLIT2\_Negative，之后单击 OK 按钮关闭窗口。

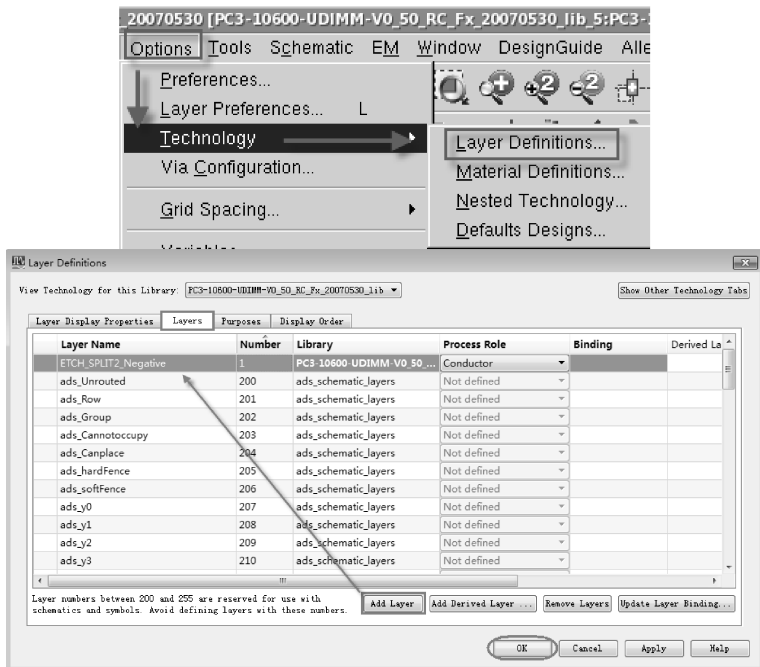


图 10-59 添加新的板层

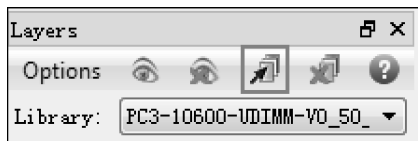


图 10-60 板层可选性/可见性快捷按钮

(2) 按下快捷键“L”，并单击“All Selectable”确保所有的板层都可以选择，如图 10-60 所示。

(3) 将 Strip 层转换为 Slot 层。如图 10-61 所示，单击版图菜单项 Adf Tools→Convert Strip <-> Slot...，如图设置参数，单击 OK 或者 Apply 按钮进行转换，只需要按任一个按钮一次就可以了。

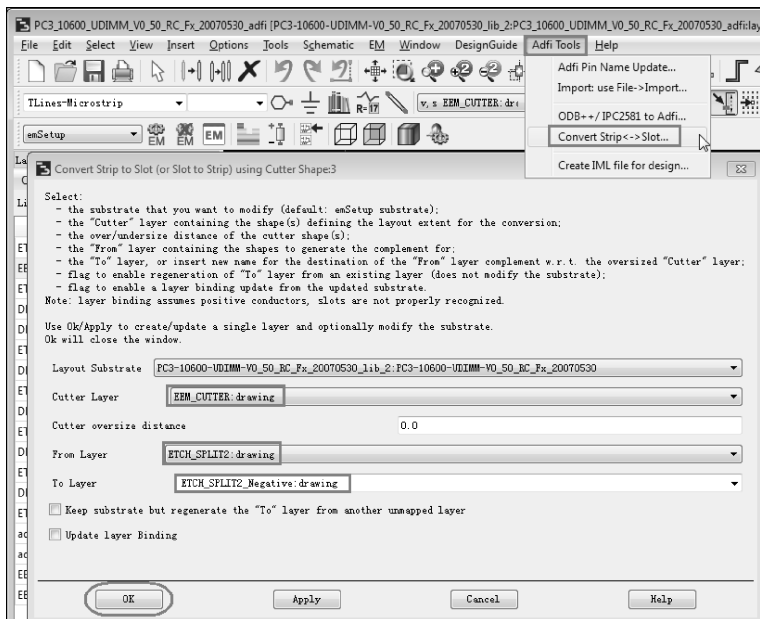


图 10-61 Strip - Slot 层转换

图 10-62 所示的是 ETCH\_SPLIT2\_NEGATIVE 层，实际上就是用 EEM\_CUTTER 减去 ETCH\_SPLIT2 层得到的结果，也就是 ETCH\_SPLIT2 反转图案。

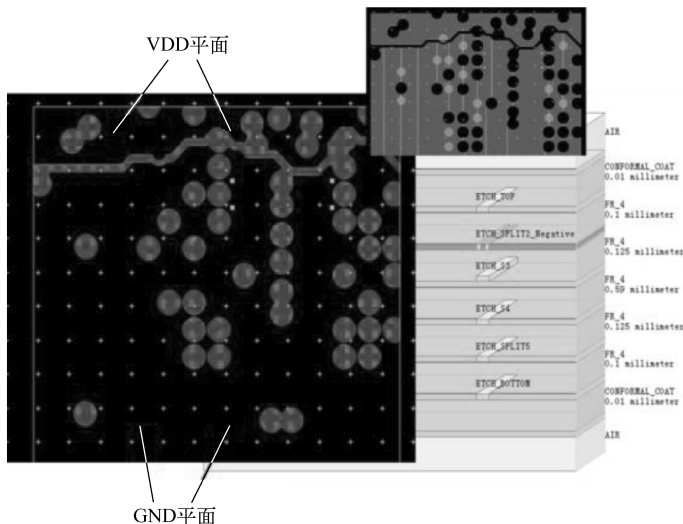


图 10-62 转换后的版图显示

另外一个发生变化的地方是在板层设置中，ETCH\_SPLIT2\_NEGATIVE 自动被映射成 Slot 层。从图 10-62 还可以看出，右上角的原来 ETCH\_SPLIT2 层略图中，VDD 网络和 GND 网络是分隔开的，而经过 Strip 层到 Slot 层的转换，空白的区域为金属走线，由于 Momentum 仿真时平面方向上的扩展性，VDD 和 GND 布线网络将会被连接在一起。

(4) 将包含电源和地平面的 Strip 层转换为 Slot 层。上面实验步骤中的 Strip 层到 Slot 层的转换会造成 VDD 网络和 GND 网络的联通，在下面的实验步骤中会将 VDD 网络和 GND 网络分割开。

首先恢复原来的介质设定。如图 10-63 所示，打开板层定义窗口，使用鼠标选中 ETCH\_SPLIT2\_Negative，并在旁边的层选择下拉框中选择 ETCH\_SPLIT2。

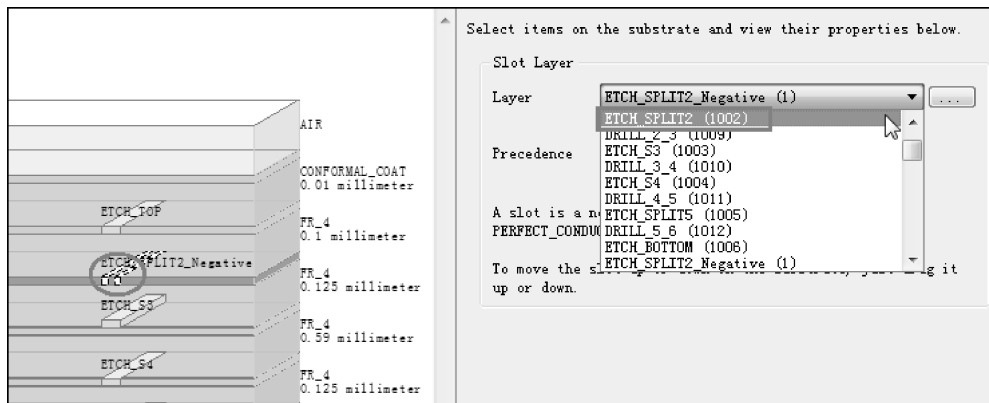


图 10-63 恢复版图 Strip 层设置

然后单击三维显示中 ETCH\_SPLIT2 层边上的灰色区域，然后在右侧单击 Strip plane 选项，如图 10-64 所示。

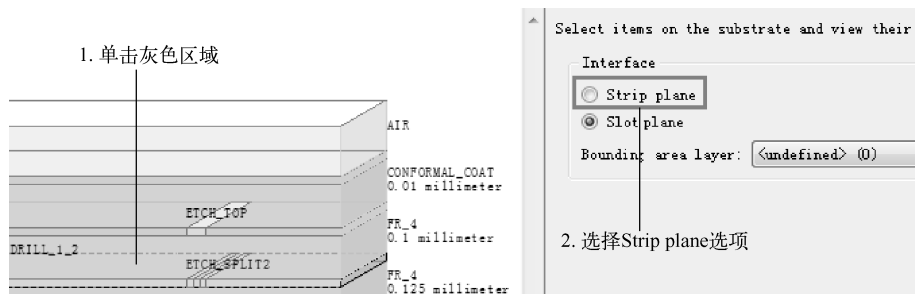


图 10-64 设置板层为 Strip 层

在弹出的对话框中单击 OK 按钮确认更改为 Strip Plane。

在版图窗口添加新的板层，命名为 ETCH\_SPLIT2\_slot（使用 ADS 版图窗口中的菜单项 Options→Technology→Layer Definitions，在弹出的窗口中单击 Add Layer 创建新的板层），并确保所有的层都是可选的（按下快捷键“L”，并单击“All Selectable”）。在版图窗口单击菜单项 Adfi Tools→Convert Strip <-> Slot...进行 Strip 层到 Slot 层的转换，如图 10-65 所示。

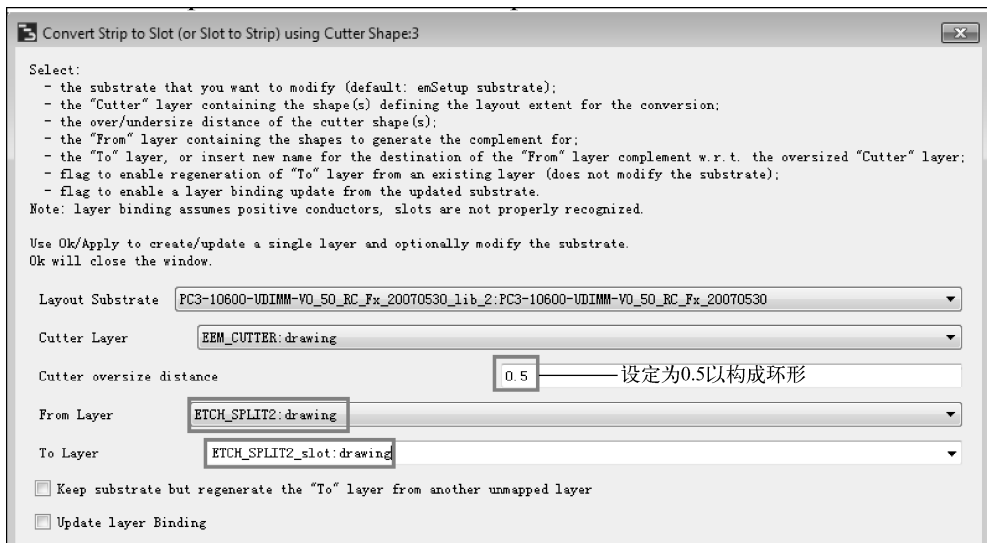


图 10-65 在进行 Strip &lt;-&gt; Slot 转换时设置间距

如图 10-65 所示，cutter oversize distance = 0.5。

可以看到，新的反转图案相比之前多了一圈环形（单击快捷键“L”，只选择 ETCH\_SPLIT2\_slot 可以显示和选择），如图 10-66 所示。

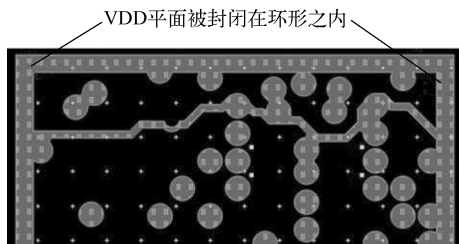


图 10-66 新转换的版图中包含封闭环形

从图 10-66 中可以看到，VDD 网络被封闭在环形之内，同时，一部分 GND 网络也被封闭在下边的环形中，这部分 GND 网络应该是和无限大地平面连接在一起的。

在此步骤中，打通 GND 网络使其连接到无限大地平面上，将 VDD 网络保持在环形中使其与无限大地平面隔离开，成为有限的电源平面。

在版图窗口，单击菜单 Select→Select All（或者使用键盘 Ctrl + A）。

单击版图菜单：Edit→Modify→Chop，之后先在图 10-67 左边的分割线处画一个矩形，之后再次使用 Edit→Modify→Chop 命令在右边的分割线画一个矩形，将原来的完整环形割断，如图 10-67、图 10-68 所示。

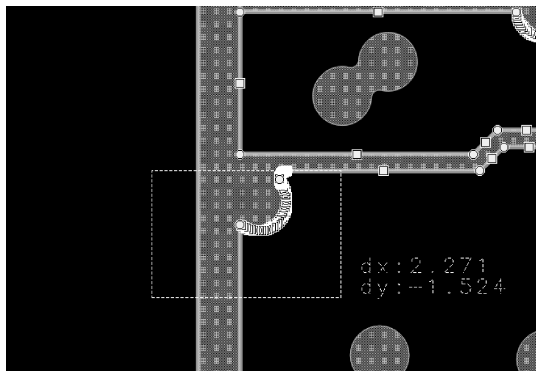


图 10-67 使用版图编辑 Chop 功能切断版图

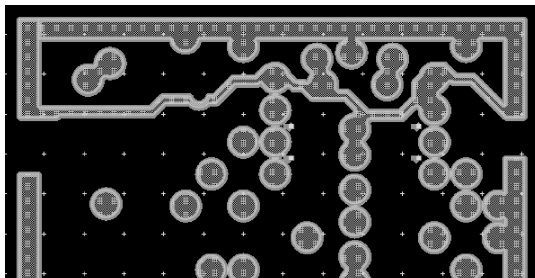


图 10-68 切断后的版图形状

之后删除下边已经割断的“环形”。上边环形之内的空白区域不会连接到无限地平面，如图 10-69 所示。

在板层设置中 ETCH\_SPLIT2\_Slot 层被自动映射成 Slot 层，如图 10-70 所示。

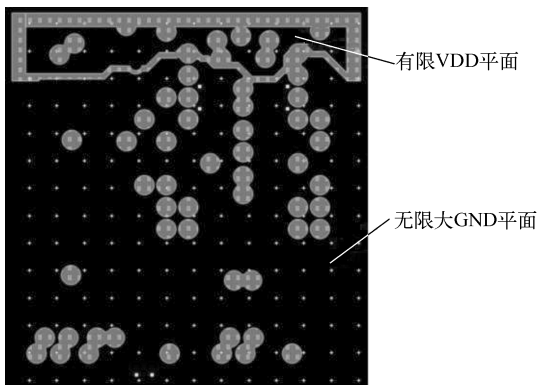


图 10-69 修改后的版图中电源网络和地网络已经隔开

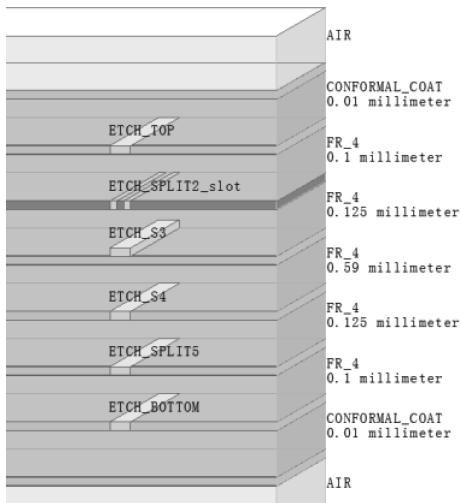


图 10-70 新的层叠设置

### 3. Ports（端口）设置

如图 10-71 所示，在 EM 设置窗口中，单击 Ports 项，可以看到版图中的端口及其设置。当激活 Auto Select、Auto Center、Auto Zoom 选项时，点中右边的端口，在版图窗口中就可以看到端口被高亮显示。

Momentum 仿真中的端口校准类型有下面几类：

- ③ TML：用来连接传输线元件。S 参数仿真结果中，激励馈线的寄生效应和馈线末端的开路效应自动被消除了。

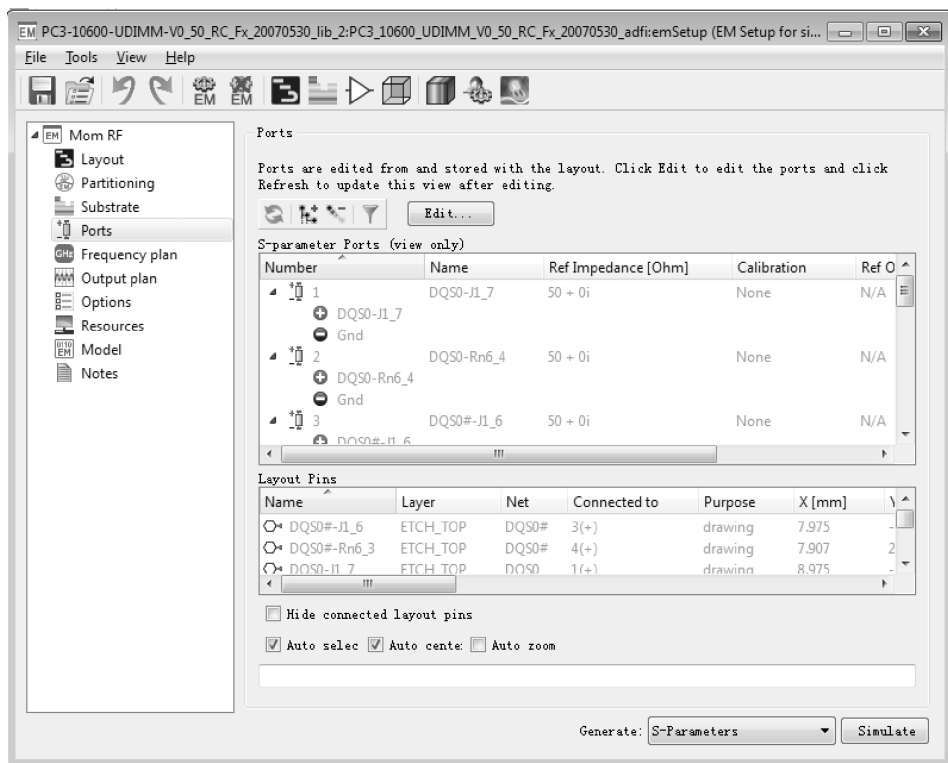


图 10-71 EM 设置窗口中的端口设置界面

☺ TML (zero length): 用来连接集总参数元件。S 参数仿真结果中, 端口的开路寄生效应自动被消除了。

☺ None: 如果不符合可选的几种端口校准类型, 就可以选择 None 选项。如果版图中的端口放在图形的内部, 端口校准类型只能选择 None。

本实例中的所有端口, 校准类型都是 None。

#### 4. Frequency Plan (频率扫描计划)

采用 Adaptive (自适应扫描) 类型, 并且将 Fstop (仿真终止频率) 改为 1GHz, Npts (最大频率扫描点数) 改为 200, 如图 10-72 所示。

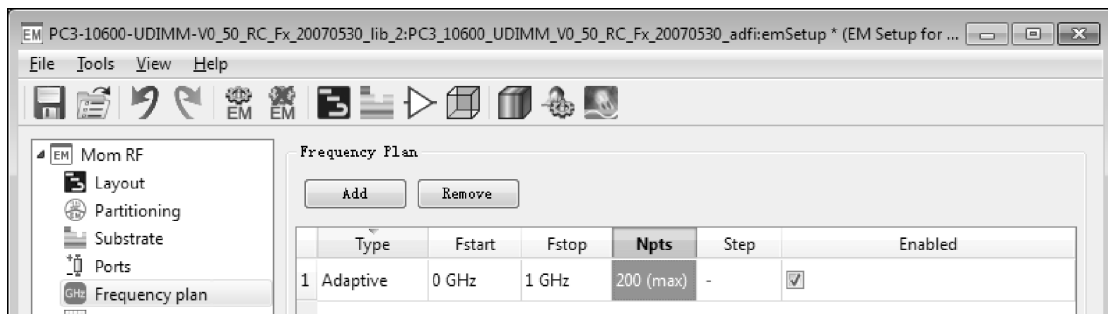


图 10-72 EM 设置中的频率扫描计划设置界面

## 5. Output plan (输出计划) (见图 10-73)

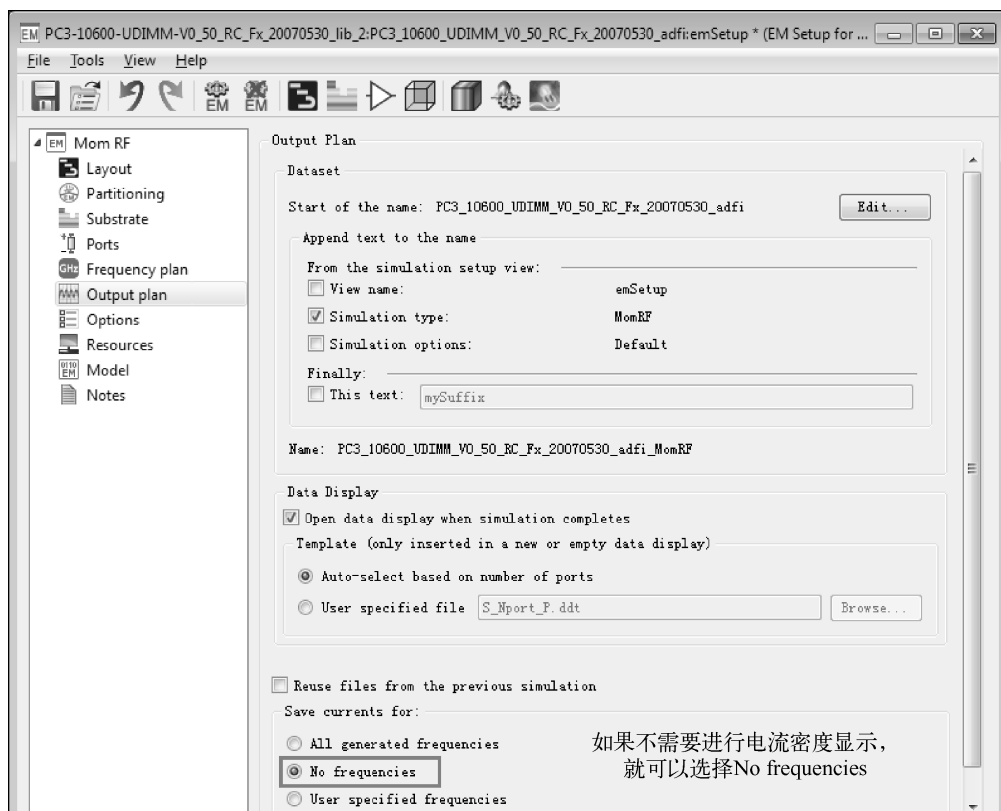


图 10-73 EM 仿真输出设置

## 6. Model (模型)

在 Model 生成的页面中，选择 Auto - create EM model when simulation is launched，如图 10-74 所示。

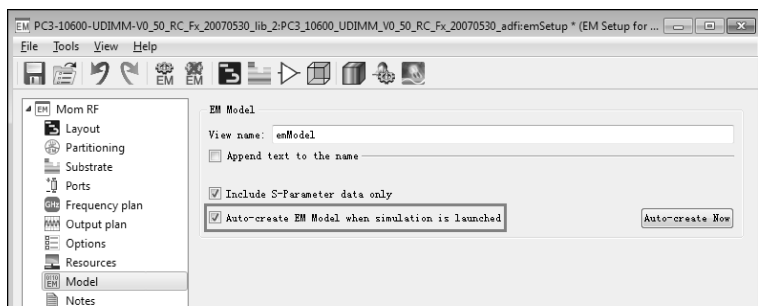



图 10-74 EM 仿真模型输出设置

## 7. 运行仿真及查看仿真结果

其他选项使用默认设置，单击快捷按钮  运行 EM 仿真。

在计算机下边的状态栏中单击 EESOF Job manager 图标，打开 EESof Job Manager，如

图 10-75 所示。



图 10-75 EM 仿真任务管理窗口

选中仿真任务名称，单击鼠标右键，选择 Open Log，如图 10-76 所示。

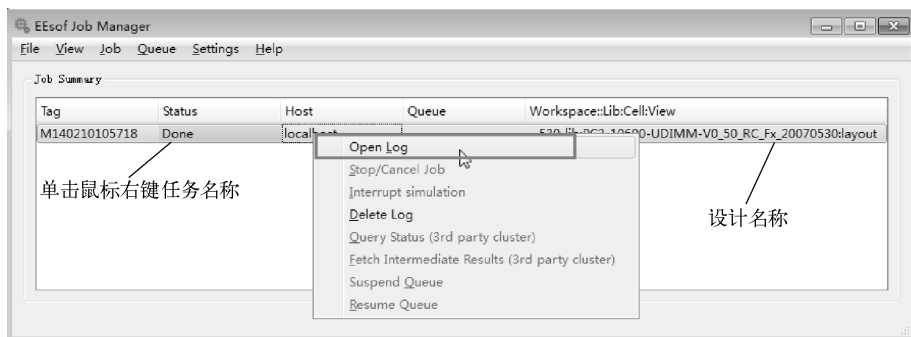


图 10-76 EM 任务管理

这时就可以看到仿真状态报告，如图 10-77 所示。

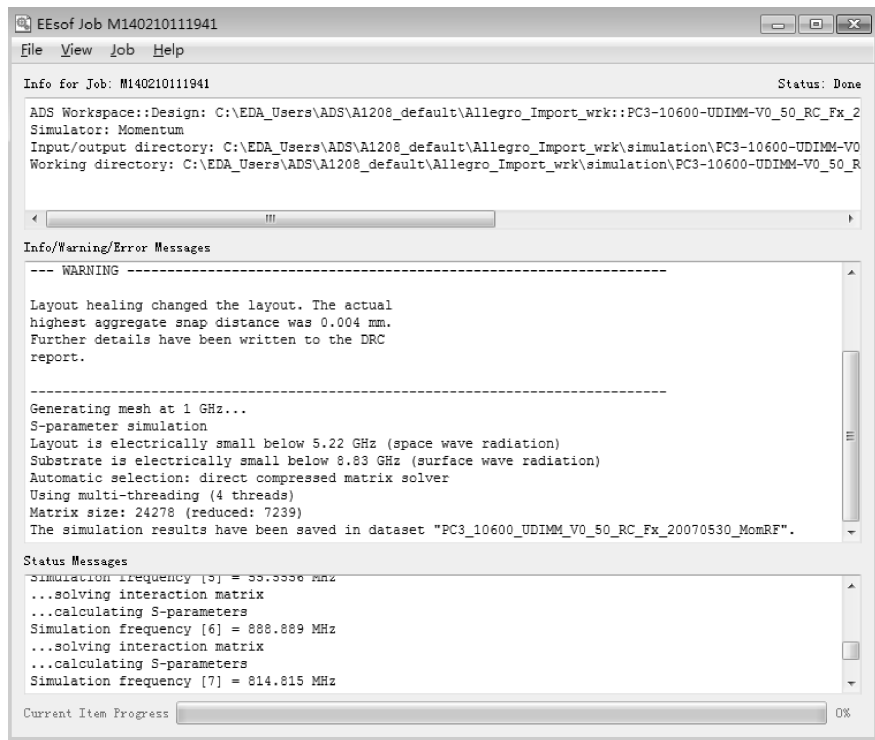


图 10-77 EM 仿真信息



从仿真报告窗口可以看到仿真过程中的所有信息，包括出错、警告、仿真时间等信息。

仿真完成后，单击版图窗口中的菜单项：

EM→Show Most Recent→S - Param Simulation summary 可以看到下面一些有用信息：

☺ 网格剖分总结信息。

☺ 求解过程中在每一个频率点上载入和求解矩阵所花费的时间（本实例中有 7 个频率点）。

☺ Momentum 自适应频率扫描达到收敛使用的频率点数（在 Frequency Plan 中设定最大仿真点数为 200）。

☺ 求解过程中消耗的内存 Physical Memory（本实例中为 1219.79MB）。

☺ User Time（用户时间）：单线程运行仿真需要的时间（本例中为 39 分 42 秒）。

☺ Elapsed Time（总体仿真时间）：多线程总体仿真时间（本例中为 18 分 17 秒）。

在 ADS 软件数据显示窗口中显示 MomentumRF 仿真结果。

Momentum 仿真结果是以 ADS 仿真结果后缀为 .ds 的格式存储的。仿真结束后会自动弹出数据显示窗口（在 EM 设置中的 output plan 中可以设置）。Momentum/Momentum RF 的默认端口阻抗是  $50\Omega$ 。

可以通过拉动滑动条改变现实的 S 参数端口号，如图 10-78 所示。

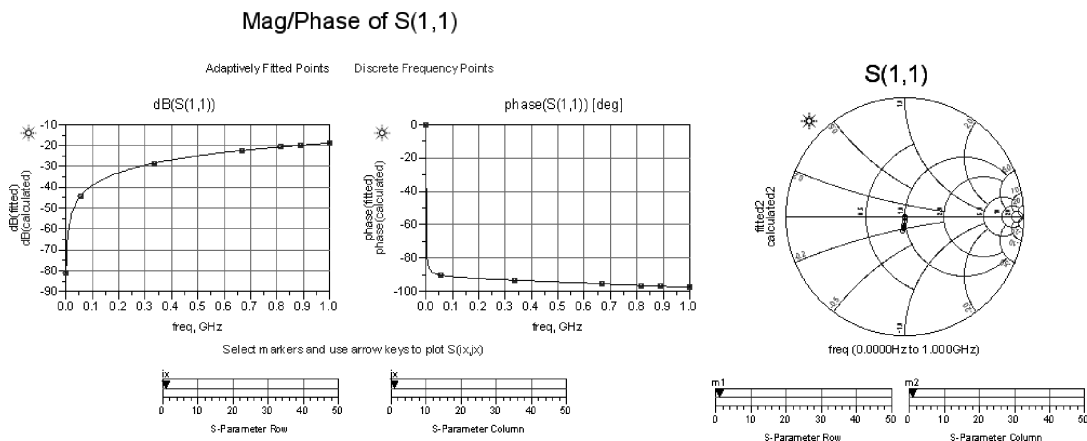


图 10-78 EM 仿真结果



## 10.6 将电磁场仿真结果代入 ADS 原理图进行仿真

### 1. 创建原理图符号

如果电路板上还有电容、电阻或者晶体管等分立器件和金属走线连接，可以在 ADS 软件的原理图设计界面中添加这些元件，并与金属走线进行连接。

回到 ADS 主窗口，单击原理图图标，建立新的原理图，命名为 DQS0\_diff，如图 10-79 所示。

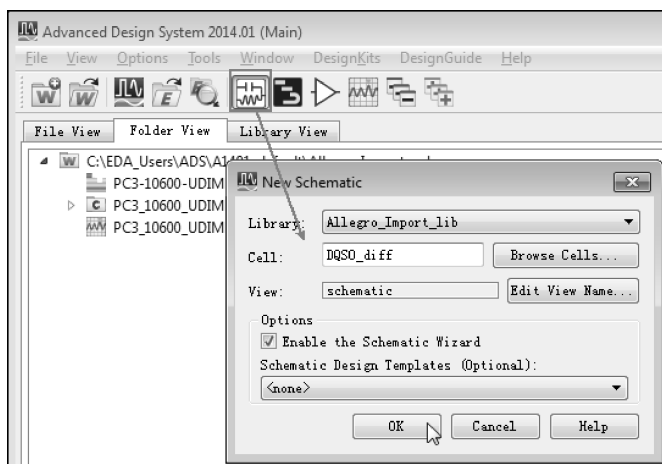



图 10-79 使用快捷图标建立新的原理图

单击 OK 按钮后，会弹出原理图设计向导，单击“Cancel”按钮关闭向导。

可以通过两种方法将 EM 仿真模型放入原理图：①在 ADS 软件主窗口中，使用鼠标点开 PC3-1060-UDIMM…单元，选中 symbol，将 symbol 拉入到原理图中；②单击原理图中的 Library（书架）图标 ，在 Component Library 中找到 Workspace Libraries，找到版图元件，拉入到原理图中，如图 10-80 所示。

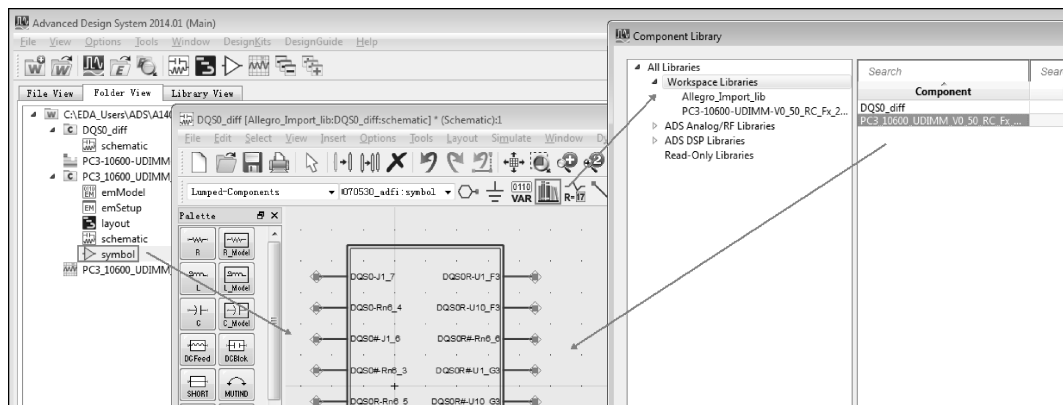


图 10-80 将已仿真好的 EM 模型加入到新原理图中

放入到原理图中的 EM 模型元件符号是简单的矩形图标，两边带有引脚及引脚名称。这样的符号（Symbol）有时在连线不太直观，可以用版图外观的符号（Layout Look-alike）进行替换：

首先回到版图窗口，在窗口左侧的 Layers 工具栏中确认只打开相关的层（如 ETCH\_SPLIT2\_Slot、ETCH\_TOP、ETCH\_S3 和 ETCH\_SPLIT5）。之后回到 ADS 主窗口中单击 symbol 元件，打开 symbol 视图。单击符号生成对话框，在弹出的页面中 Source view 下拉选项中选择 Layout，并选择 Look-alike，可以通过 symbol size 参数的设置调整生成的符号视图在原理图中的大小，如图 10-81 所示。

单击 OK 按钮结束编辑，就会生成新的符号，同时之前的原理图也会更新，如图 10-82 所示。

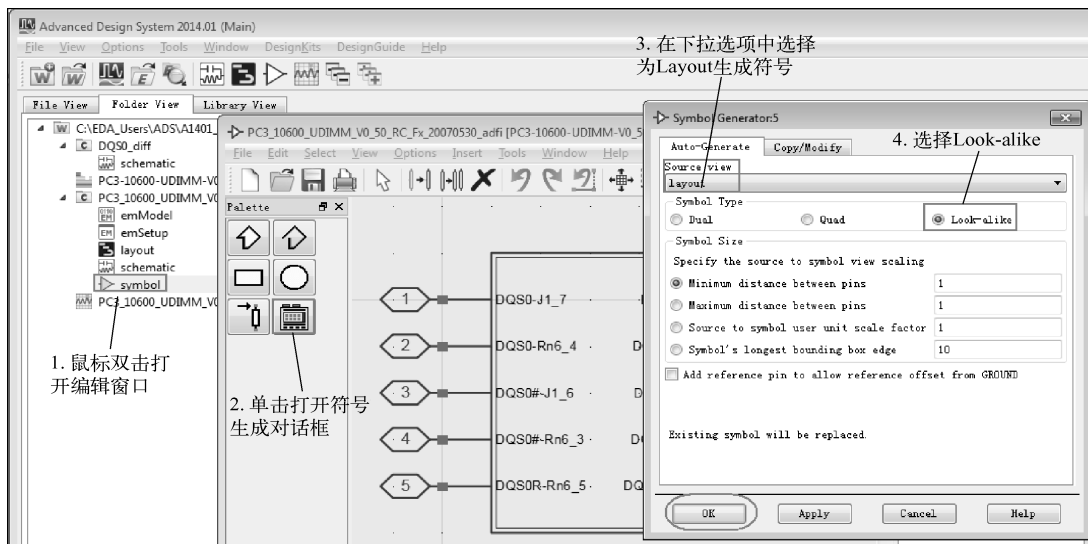


图 10-81 元件符号编辑界面

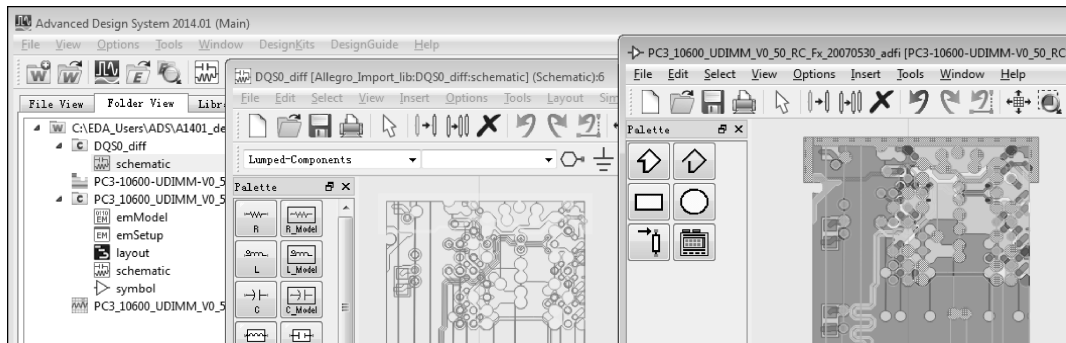


图 10-82 具有版图外观的符号

可以看到符号编辑界面，符号是彩色的，而在原理图中，版图符号大多都是灰色的，各层版图都叠加在一起，这是为什么呢，如何修正？

版图元件没有颜色，是因为原理图的单元（cell）和版图单元（cell）分别属于不同的库（Libraries）。

如图 10-83 所示，在本实例中单元 PC3 - 10600 - UDIMM - V0\_50\_RC\_Fx\_20070530\_adfi 的版图实例（Layout view）是在 PC3 - 10600 - UDIMM - V0\_50\_RC\_Fx\_20070530\_lib 的库之下，其中还包括 ADS 板层设置、原理图层设置以及导入的 Allegro 板层设置。

而使用默认设置建立的单元 DQS0\_diff 原理图是在 Allegro\_import\_lib 库之下，其中只包括默认的 ADS 版图和原理图板层设置，并不包含新导入的 Allegro 板层设置。

改正的办法就是在创建 DQS0\_diff 单元时选择建立在 PC3 - 10600 - UDIMM - V0\_50\_RC\_Fx\_20070530\_lib 库之下或者将已经建好的 DQS0\_diff 单元移动到 PC3 - 10600 - UDIMM - V0\_50\_RC\_Fx\_20070530\_lib 库之下。本实例中使用移动功能，首先回到原理图窗口保存原理图，并关闭所有打开的原理图窗口、符号编辑窗口，之后在主窗口的 Library View 栏下选中 test\_sparam 单元，单击鼠标右键，选择 Move Cell...，在弹出的窗口中选择目标库 PC3 - 10600 - UDIMM - V0\_50\_RC\_Fx\_20070530\_lib，并修改名称为 DQS0\_diff，如图 10-84 所示，设定完成后单击 OK 按钮结束。

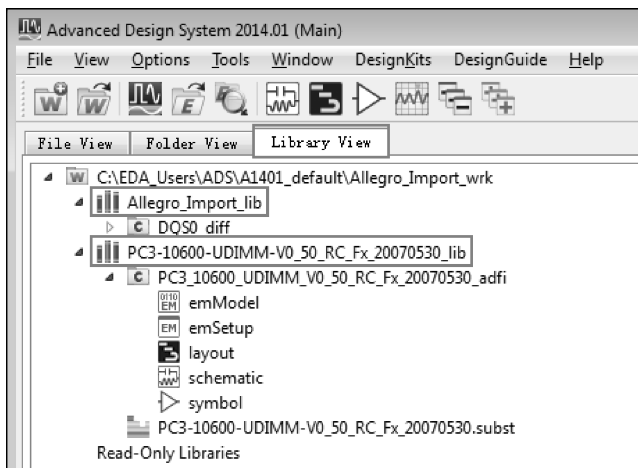


图 10-83 ADS 软件的库管理界面

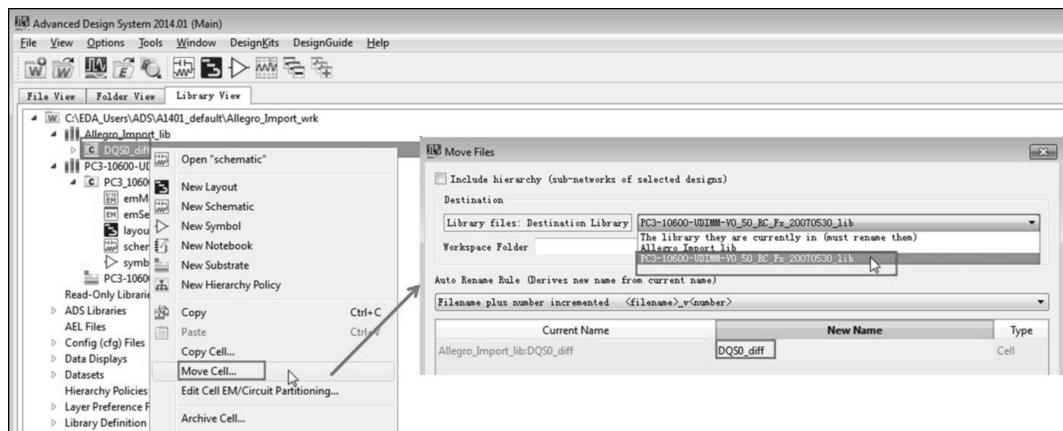


图 10-84 设计单元在两个库之间移动

这样，就在 PC3 - 10600 - UDIMM - V0\_50\_RC\_Fx\_20070530\_lib 库之下建立了名为 DQS0\_diff 的单元，双击打开的 DQS0\_diff 原理图单元，打开原理图窗口，当前的视图就不再是灰色的了，如图 10-85 所示。

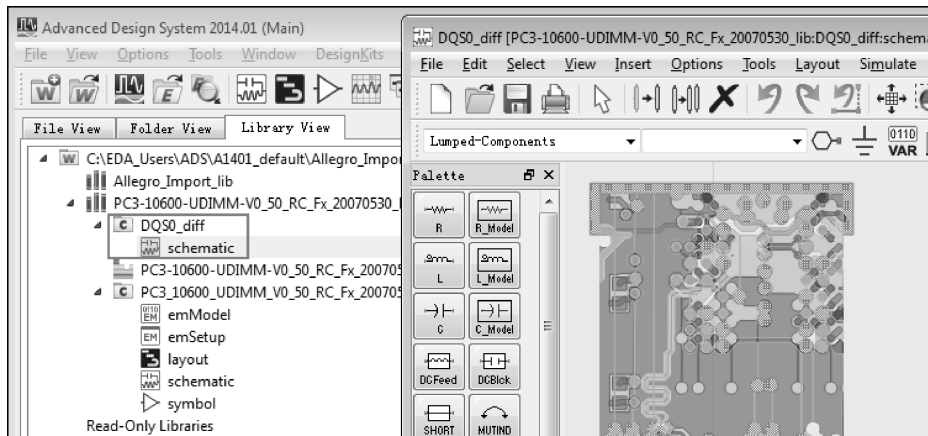


图 10-85 在同一个库中时，版图元件的符号显示

## 2. 插入 $0\Omega$ 电阻并创建 4 端口符号视图

在原理图下方加入 Port1、Port3，并使用  $0\Omega$  电阻短接断开的走线，在上方加入 Port2、Port4，如图 10-86 所示。

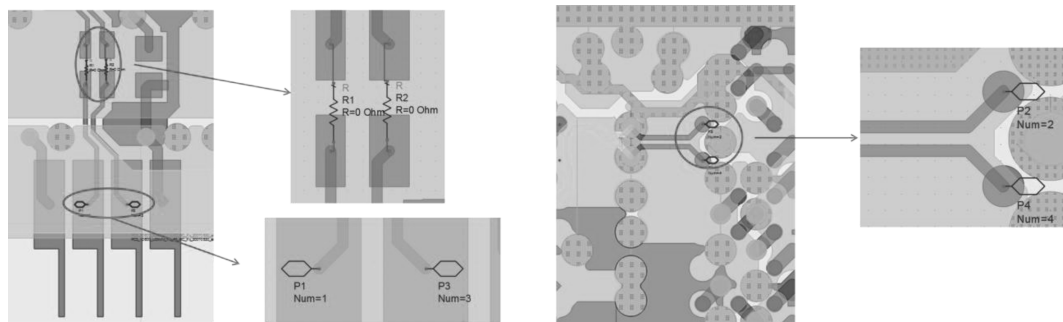



图 10-86 在版图元件上进行连线

在原理图窗口，单击菜单栏上的  图标，再使用鼠标单击版图元件，选择进行仿真的 View，如图 10-87 所示。

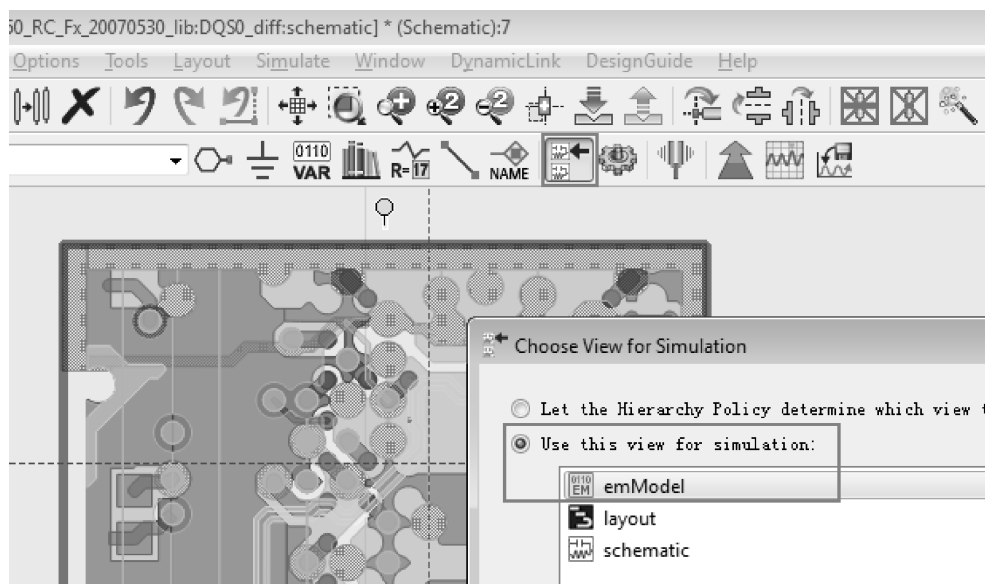


图 10-87 仿真模型选择界面

完成后，回到 ADS 主窗口，使用鼠标选中 DQS0\_diff 单元，单击鼠标右键，选择 New Symbol，建立符号视图，如图 10-88、图 10-89 所示。

## 3. 运行 S 参数仿真

新建 test\_sparam 原理图，插入新建的 DQS0\_diff 子电路，并加入端口及 S 参数控制器，设置频率扫描范围，运行 S 参数仿真，如图 10-90 所示。

仿真完成后，可以在弹出的数据显示窗口中加入下面的显示，如图 10-91 所示。

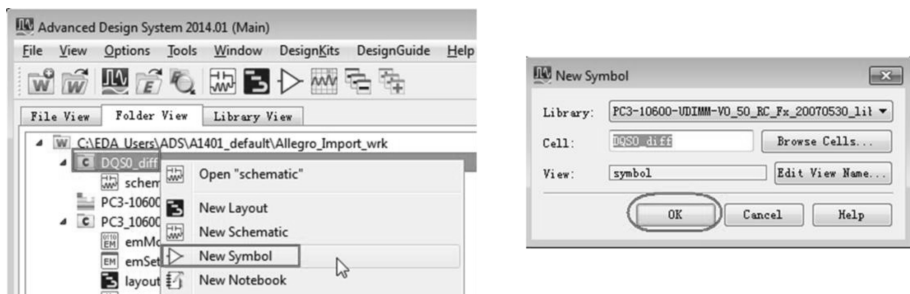


图 10-88 建立新的符号

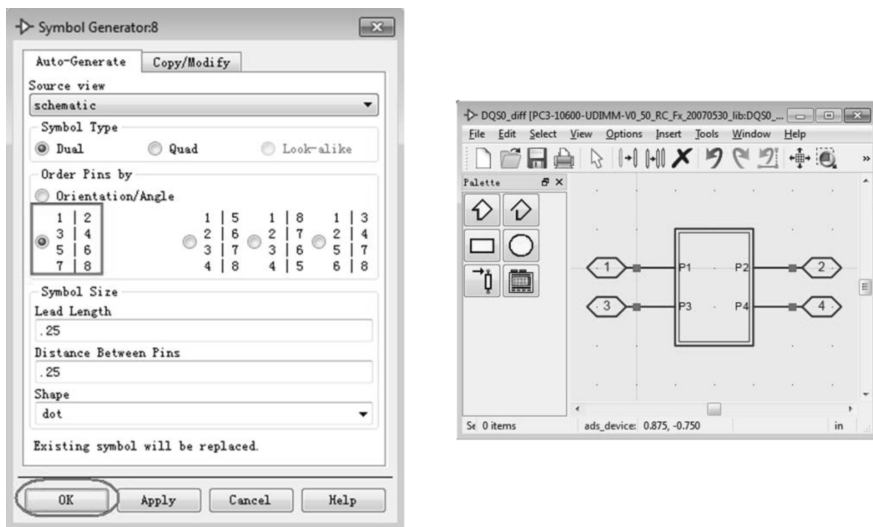


图 10-89 建立新的符号界面

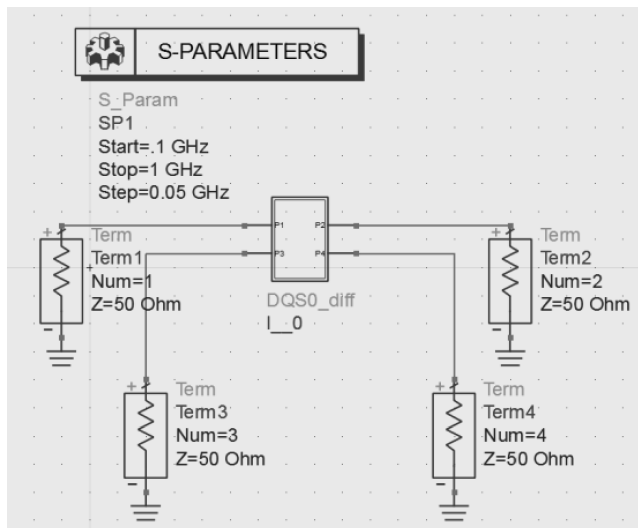


图 10-90 插入新建立的符号运行 S 参数仿真

新建 test\_diff\_sparam 原理图，在原理图中加入巴伦元件（System - Passive 元件面板）对差分走线进行仿真，注意修改 Term 元件的阻抗，如图 10-92 所示。

差分仿真结果如图 10-93 所示。

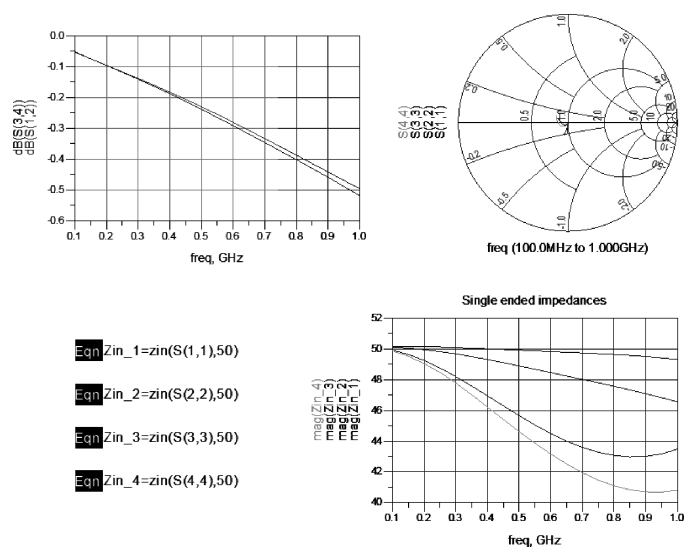


图 10-91 S 参数仿真结果

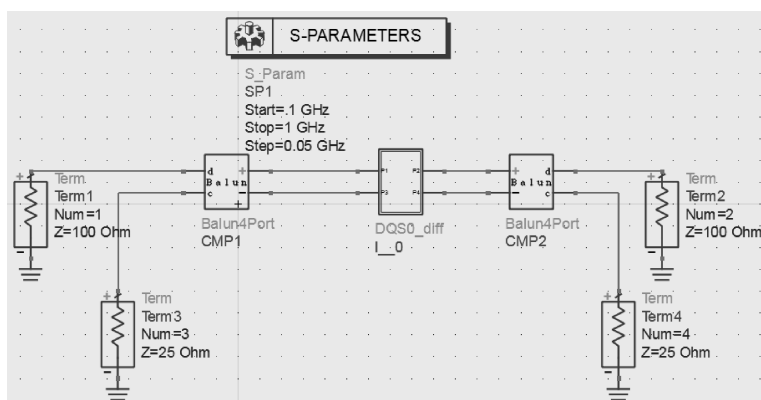


图 10-92 差分 S 参数仿真模板

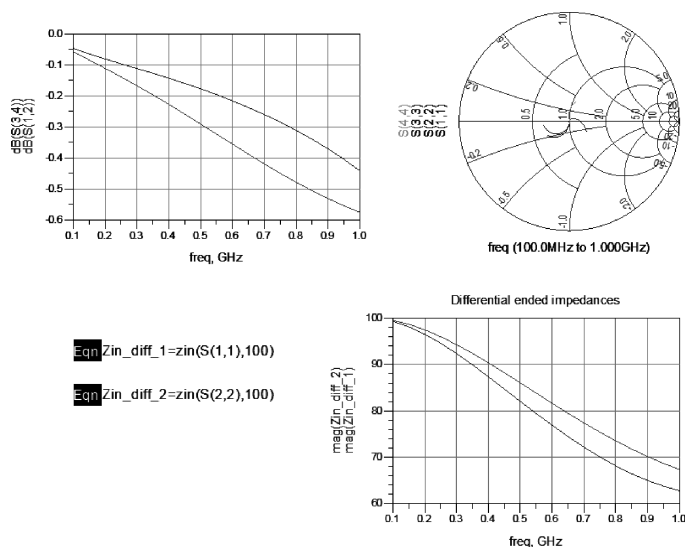


图 10-93 差分 S 参数仿真结果

#### 4. 运行瞬态仿真

新建 test\_trans 原理图，在原理图中加入时域源（Vf\_Square 元件在 Sources – Freq Domain 面板中），进行时域仿真，如图 10-94 所示。

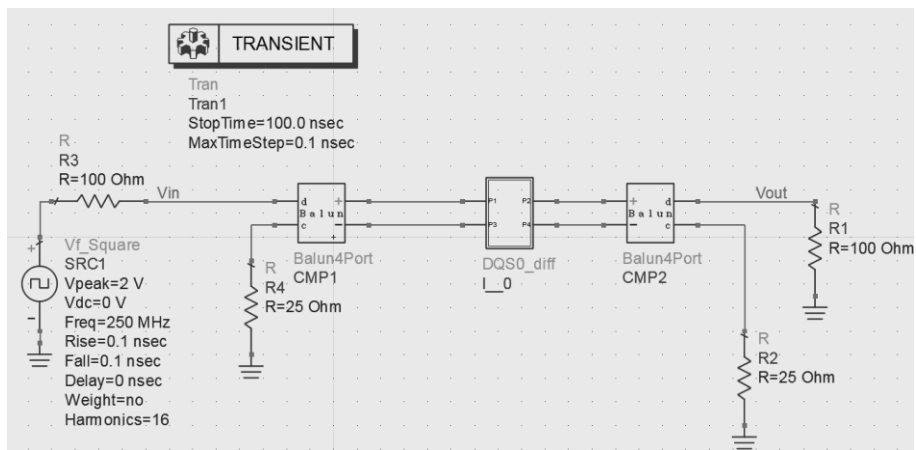


图 10-94 差分激励瞬态仿真模板

仿真结果如图 10-95 所示。

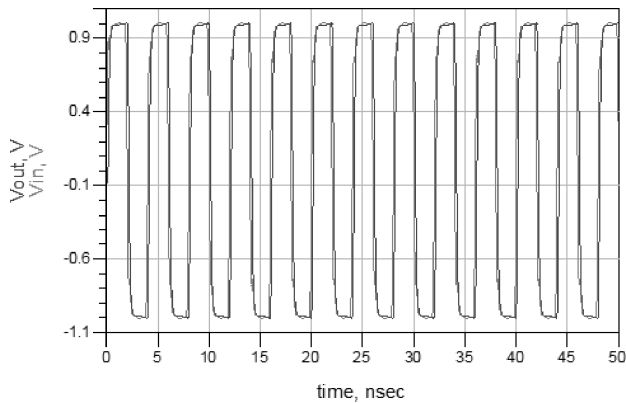


图 10-95 瞬态仿真结果

#### 【总结】

本实例介绍了将 Cadence Allegro 版图中的关键走线网络导入到 ADS 软件中进行电磁场仿真的过程。详细介绍了专用插件的安装、走线网络提取过程以及导入 ADS 软件后进行电磁场仿真所需的设置及注意事项。在电磁场仿真完成后，可以将仿真结果用于频域、时域的仿真测试。



# 实例 11 DDR4 仿真及一致性测试

## 【目的】

本实例主要介绍如何使用 ADS 最新发布的 DDR4 Test Bench，快速搭建 DDR4 拓扑结构进行仿真，并将仿真结果直接导入到 Keysight 公司最新脱机版 Infiniium 中进行自动的一致性测量并生成测试报告的流程。

## 【背景知识】

### 1. DDR 一致性测试简介

DDR 是双倍数据率的 SDRAM 内存接口，JEDEC（固态技术协会）于 2000 年发布了第一代 DDR1 的设计规范，经过几代技术的发展，如今 DDR3 已经成为市场上的主流，其数据速率最高达到 2333Mbps，而最新的 DDR4 的数据速率更能达到了 3200Mbps。随着数据速率的不断升高，电子工程师在设计时所面临的信号完整性挑战也越来越大。原来遵循的等线长设计规则已经不能保证设计的成功，因此更需要仿真来分析设计的瓶颈所在并在设计初期尽早地发现问题。

而对于 DDR3/4 的仿真来讲，最大的挑战莫过于 DDR 的拓扑结构往往非常复杂，而且 JEDEC 规范所定义的一致性测试项目多达上百项之多，这使得手动测量所有项目并生成测试报告非常耗时耗力。因此如何高效地搭建 DDR 拓扑结构进行仿真以及自动地进行一致性测量成为 DDR 仿真的最大挑战，如图 11-1 所示。

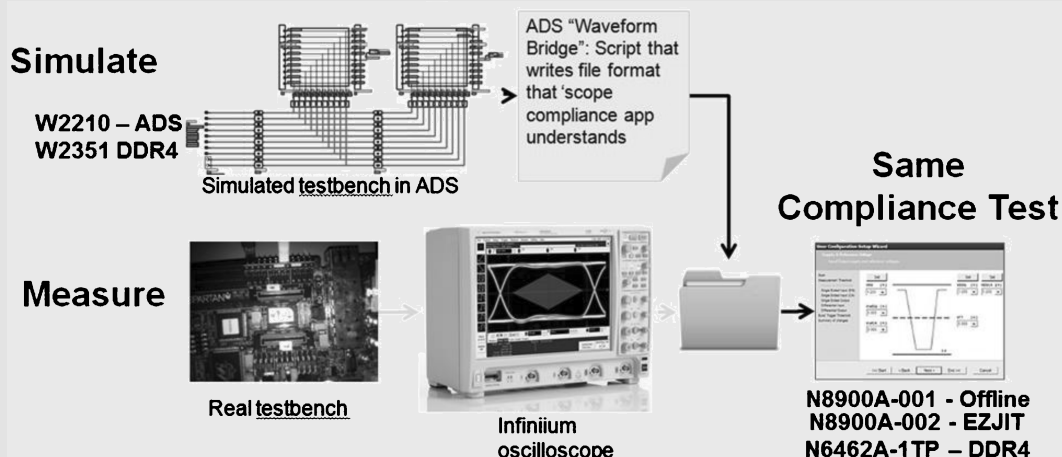


图 11-1 仿真和实测采用相同的一致性测试软件

ADS2014.01 中最新发布的 DDR4 Test Bench 完全针对上面提到的两个仿真测试难点，可以帮助用户高效精准地完成整个 DDR4 的仿真及一致性测试工作。而其一致性测试采用和 Keysight Infiniium 90000 示波器中完全相同的测试应用软件，最大限度地保证了仿

真和测量在算法应用上的一致性。为用户的设计提供最可靠的一致性测试结果。

## 2. DDR4 拓扑结构

传统的 DDR1 和 DDR2 的地址、控制、时钟总线普遍采用对称的 T 型拓扑，此种拓扑结构要求时钟等信号的走线要“等长”，因此对于布线的要求很高，且由于分支较多造成走线的阻抗变化，随着数据速率的增高更容易造成数据完整性问题。因此 DDR3 和 DDR4 的地址、时钟等总线普遍采用 Fly-by 拓扑结构，该结构为一种改良的菊花链结构。Fly-by 拓扑与 T 型拓扑的对比如图 11-2 所示。Fly-by 拓扑中的时钟、地址等总线从存储器控制器开始以串行的方式连接到每个 DRAM 单元，通过减少分支数量和分支长度来改善信号完整性。不过由于时钟信号到每个单元的时间延迟不同，DDR3/DDR4 的控制芯片一般都采用 Read/Write Leveling 技术进行时序补偿。

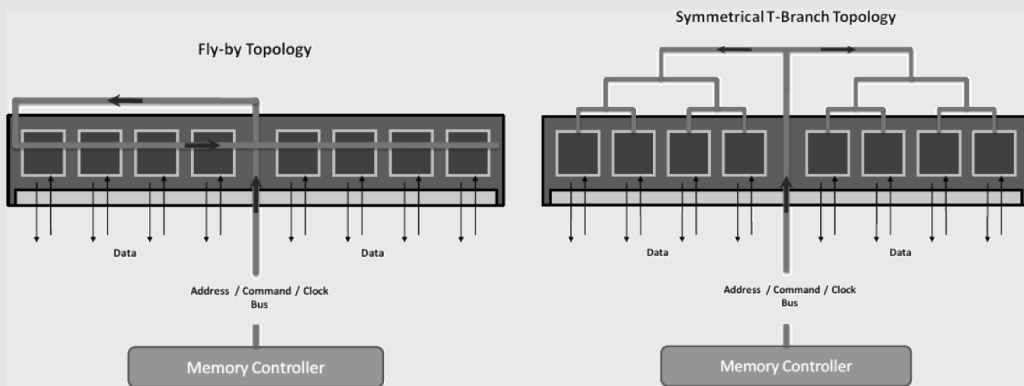


图 11-2 Fly-by 拓扑与 T 型拓扑的对比

除拓扑结构外，控制芯片所连接的 DIMM 个数以及 Rank 个数也多种多样，在仿真中不可能覆盖所有的拓扑结构，因此仿真时我们往往选取典型的走线结构来进行。

## 3. DDR4 Test Bench 仿真模板简介

(1) 打开实例下载资源中的 DDR4\_Compliance\_Test\_Bench\_wrk.7zads 项目文件。这个项目文件包含图 11-3 所示的几个子文件夹，其中“\_Simulation test benches”包含所有读、写、CA 等顶层仿真的原理图，而下面的文件夹则包括被调用的 driver、receiver、DIMM、PCB 模型等。

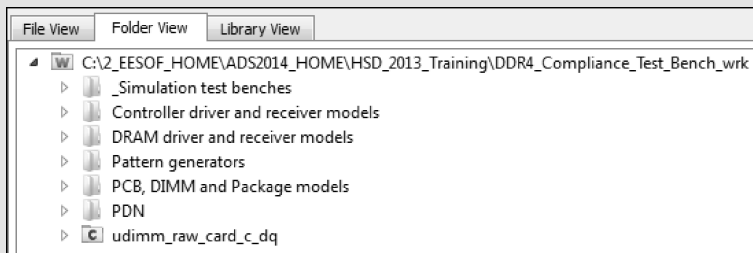


图 11-3 打开 ADS 项目文件

(2) 如图 11-4 所示, 打开 “\_Simulation test benches” 文件夹, 里面包含如图 11-5 所示的仿真模板。

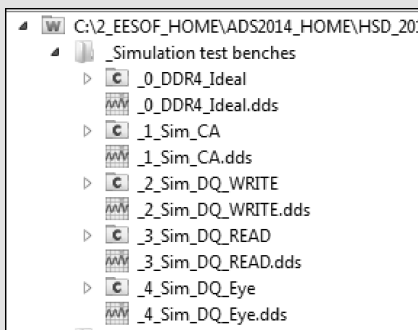


图 11-4 打开 “\_Simulation test benches” 文件夹

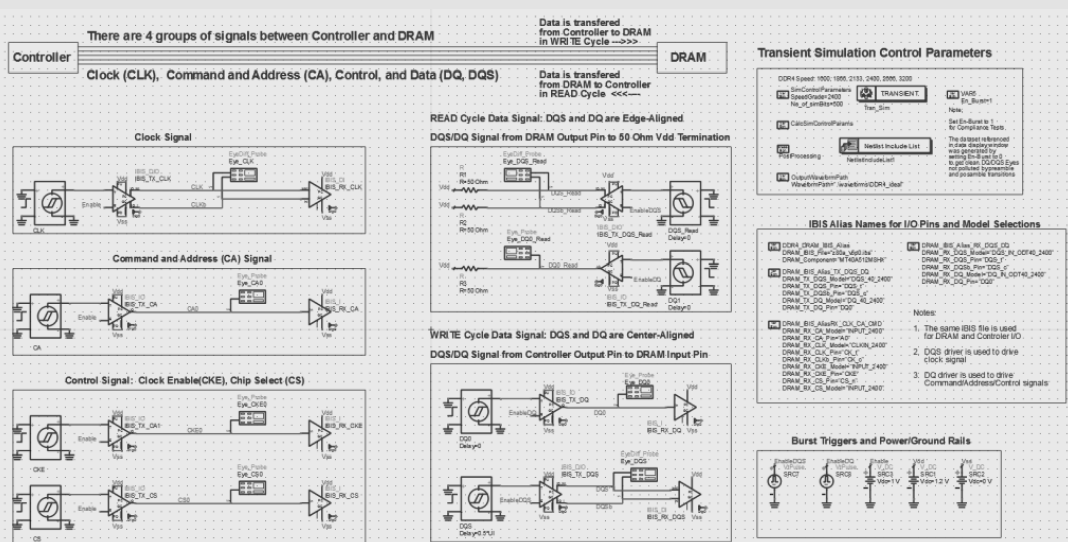


图 11-5 ADS 中的仿真模板

其中 \_0\_DDR4\_Ideal 为地址、时钟、DQ、DQS 等仿真的最基本设置模板, 需要做前仿时用户可以参照这些设置加入自己的通道模型进行仿真, 具体设置不在此一一赘述。除此之外, 这个文件夹中还包含地址控制总线 (\_1\_Sim\_CA)、读数据 (2\_Sim\_DQ\_READ)、写数据 (3\_Sim\_DQ\_WRITE) 等仿真模板。在接下来的内容中我们会详细讨论具体的仿真设置。

#### 4. CA 仿真

(1) 打开 “\_1\_Sim\_CA” 原理图, 如图 11-6 所示。此仿真模板基于 6 根单端 CA 线 (CA0 ~ CA5)、一根单端片选线 (CS0) 和一对差分时钟线 (+/- CLK/CLKb) 的拓扑结构开发。因为 ADS 支持层级化的设计结构, 因此整个模板包含 4 个顶层模块, 分别为码型发生单元 (PatternGen\_CA\_Continuous)、控制芯片单元 (Controller\_CA\_Driver)、PCB 通道模型 (PCB\_CA\_Trace) 和 DRAM 接收单元 (DRAM\_CA\_Receiver)。

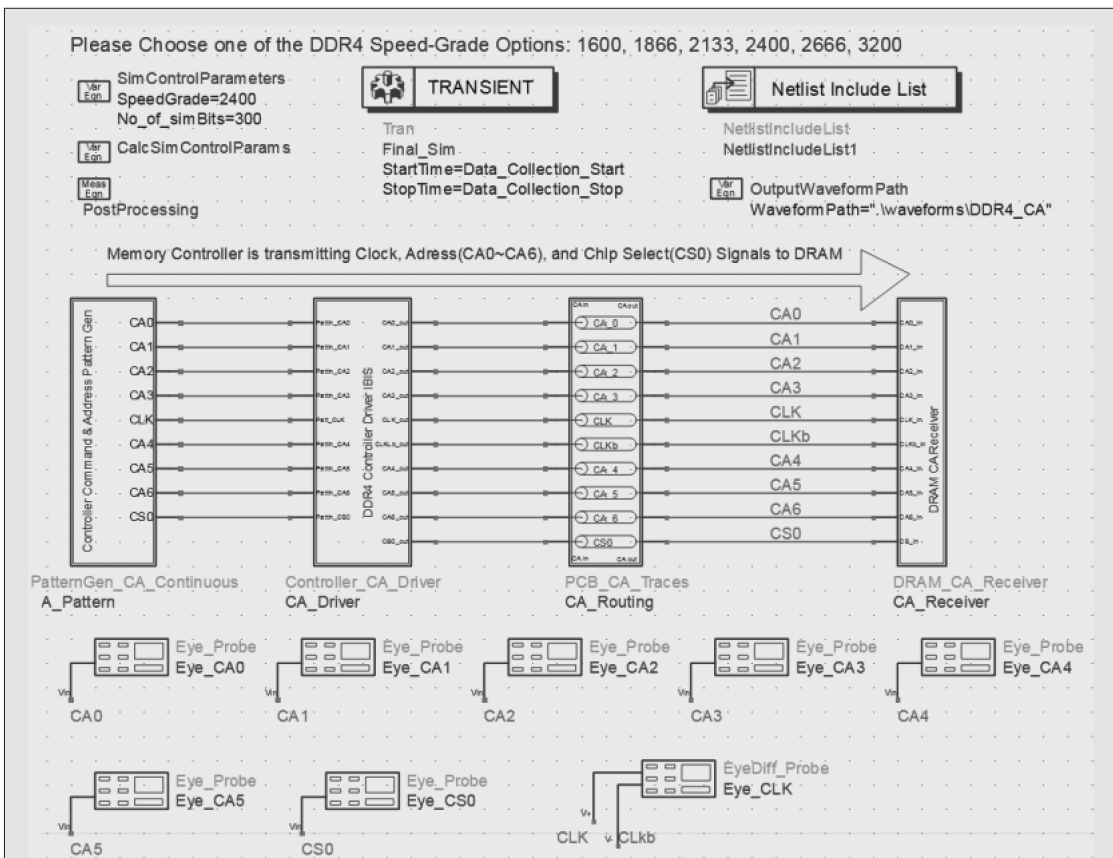


图 11-6 “\_1\_Sim\_CA” 原理图

(2) 选中“PatternGen\_CA\_Continuous”并“Push Into”，可以看到仿真的激励码型是通过一组 VtPRBS 源加载的。所有源的定义都通过变量定义，用户只需要在顶层原理图中定义 SpeedGrade，所有相应的码型参数就会被自动计算并设置好，如图 11-7 所示。

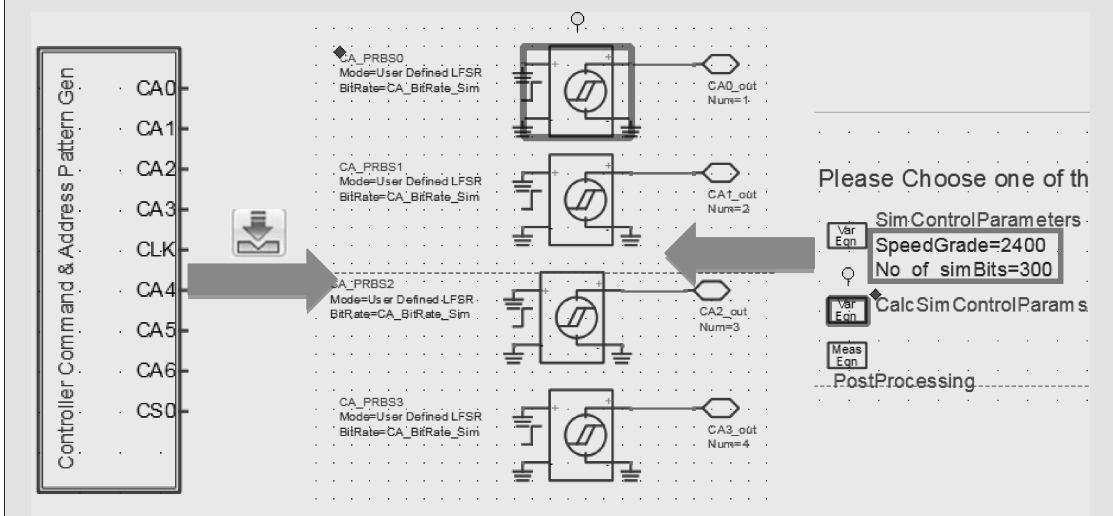


图 11-7 激励源的设置

(3) 如果需要更改码型设置, 则可以双击 VtPRBS 模块, 可以在弹出的“Edit Instance Parameters”窗口中做修改, 如图 11-8 所示。例如, 可以选取不同的 User Defined LFSR、Bit File 码型类别等。

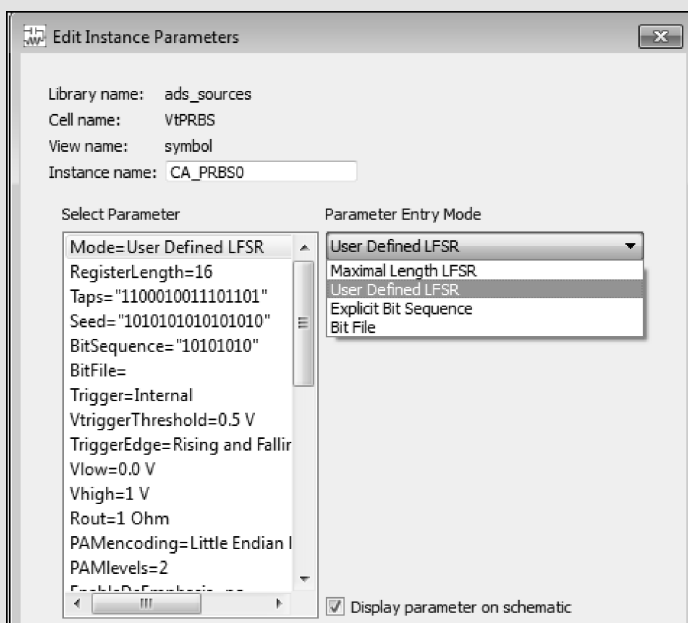


图 11-8 PRBS 源的设置

(4) “Controller\_CA\_Driver”单元中定义的控制芯片的模型, 对于 DDR 的仿真一般需要两个 IBIS 模型, 分别表征控制芯片及 DRAM 芯片的各个端口。对于 IBIS 模型的具体设置可以“Push Into”这个单元查看, ADS 中针对不同的端口类型有不同的 IBIS 模块与之对应, 如 IBIS\_IO (输入输出端口)、IBIS\_I (输入端口) 等, 双击每个模块并进入“Aliases”标签页, 可以看到所有的 IBIS 模型定义也都是通过变量来统一控制的, 这主要为了方便统一修改或替换所有端口的 IBIS 模型, 如图 11-9 所示。

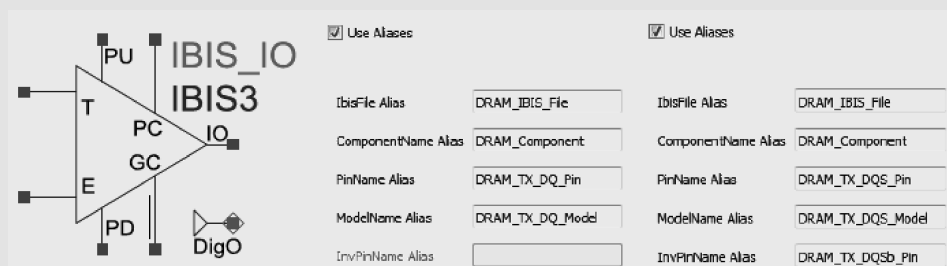


图 11-9 IBIS 元件的 Aliases 选项

如果用户需要替换自己的 IBIS 模型, 只需要根据新 IBIS 模型的参数修改相应的变量定义即可, 这样所有管脚的 IBIS 模型也会统一替换, 如图 11-10 所示。

(5) 有关 DRAM 接收单元 (DRAM\_CA\_Receiver), 其设置也是和 Controller Driver 单元一致的, 用户可以自行设置 IBIS 模型变量来配置模型。

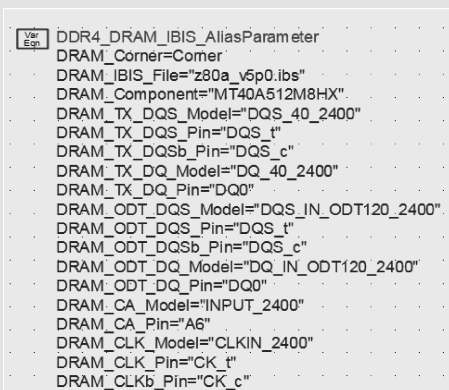


图 11-10 使用变量控制 IBIS 模型的选择

cell 名, 然后将其拖曳到顶层原理图中替换原有的“PCB\_CA\_Traces”单元即可, 如图 11-12 所示。

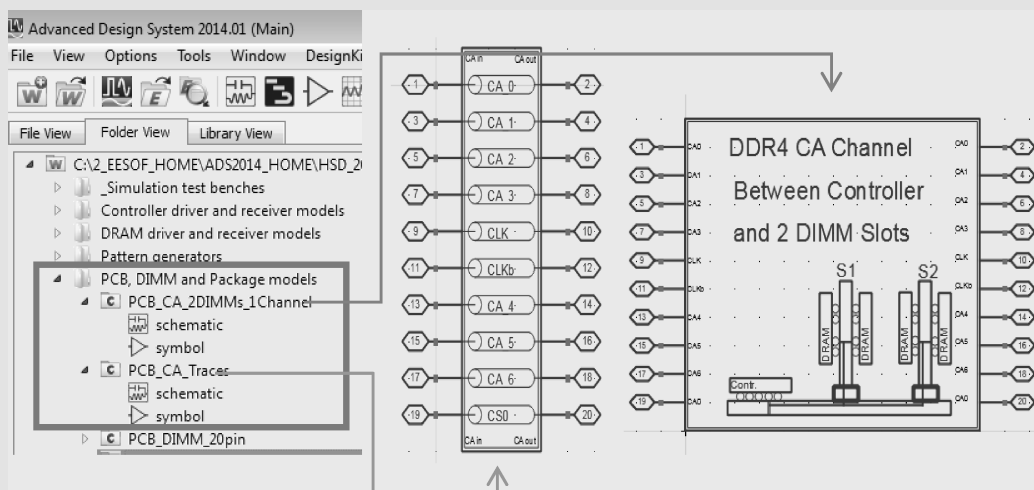


图 11-11 PCB 通道模型

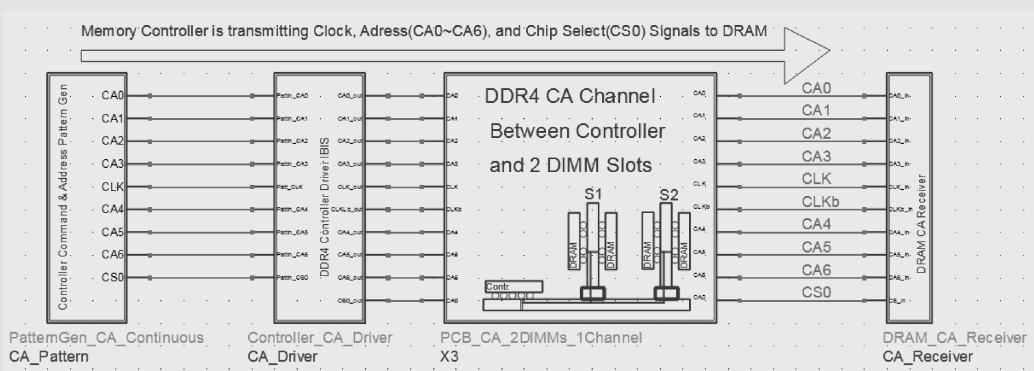



图 11-12 更改 PCB 通道模型

(8) 维持原有的 PCB\_CA\_Traces 拓扑结构并单击  按钮进行仿真, 仿真后数据显示窗口会自动弹出, 可以观测不同信号的眼图, 如图 11-13 所示。

(6) 而 PCB 通道模型 (PCB\_CA\_Trace) 则是仿真的重点, 如图 11-11 所示。不同的 PCB 和拓扑结构对应的模型不同, 在此 Test-Bench 中我们准备了两套 PCB Trace 模板, 分别是在 PCB 中 DIMM and Package models 子文件夹中的 PCB\_CA\_2DIMMs\_1Channel 和 PCB\_CA\_Traces, 前者为典型的双 DIMM 结构, 后者则是非常简单的传输线模型。

(7) 在 CA 仿真的模板中采用的是简单的 PCB\_CA\_Traces 结构, 如果想要替换, 只需

用鼠标点中“PCV\_CA\_2DIMMs\_1Channel”的



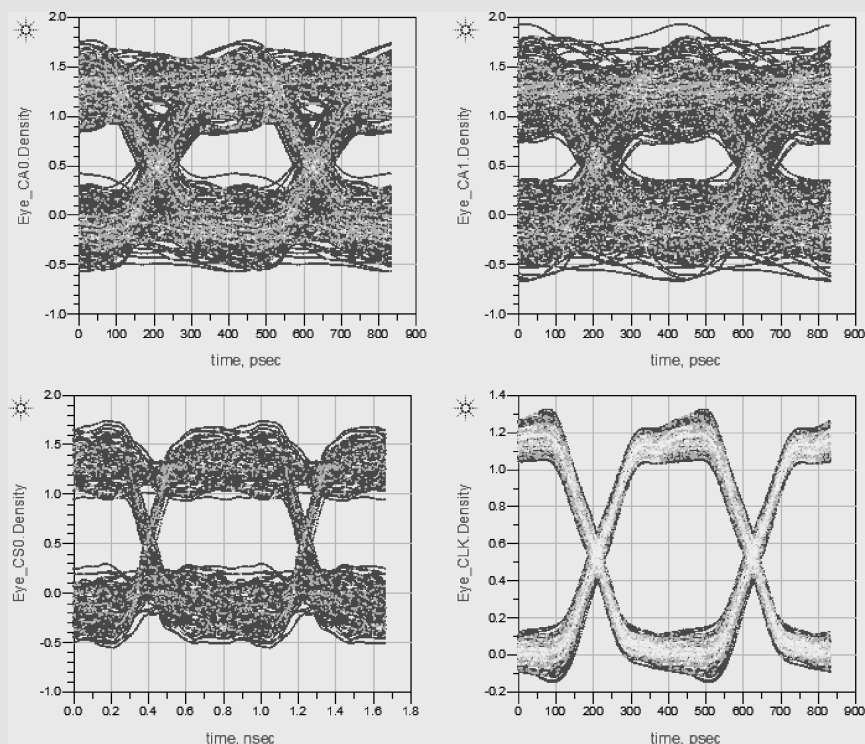


图 11-13 仿真的眼图结果

(9) 仿真结束后，模板可以自动将所有的波形转换成 .h5 格式并存储在这个项目文件的 data\waveforms\DDR4\_CA 文件夹中，如图 11-14 所示。.h5 文件格式可以直接导入到 Infiniium 中进行一致性测试分析。具体步骤将在后面的实验步骤中详述。

C:\Users\Default\DDR4_Compliance_Test_Bench_wrk\data\waveforms\DDR4_CA				
Include in library ▾ Share with ▾ Burn New folder				
	Name	Date modified	Type	Size
ites	CA0.h5	7/26/2014 11:26 AM	NCSA HDFView	17 KB
ktop	CA1.h5	7/26/2014 11:26 AM	NCSA HDFView	17 KB
ent Places	CA2.h5	7/26/2014 11:26 AM	NCSA HDFView	17 KB
NG,JIAN (A-Americas,exl)	CA3.h5	7/26/2014 11:26 AM	NCSA HDFView	17 KB
ies	CA4.h5	7/26/2014 11:26 AM	NCSA HDFView	17 KB
tuments	CA5.h5	7/26/2014 11:26 AM	NCSA HDFView	17 KB
sic	CLK.h5	7/26/2014 11:26 AM	NCSA HDFView	17 KB
tures	CLK_Diff.h5	7/26/2014 11:26 AM	NCSA HDFView	17 KB
eos	CLKb.h5	7/26/2014 11:26 AM	NCSA HDFView	17 KB
	CS0.h5	7/26/2014 11:26 AM	NCSA HDFView	17 KB

图 11-14 仿真波形保存为 .h5 文件

## 5. Sim\_DQ\_WRITE 仿真

(1) 由于对于控制芯片和 DRAM 芯片来说，其作为发射端和接收端所采用的 IBIS 模型的偏置条件和设置不同，同时在 JEDEC 标准中对于读和写的时序关系判定标准也不同，因此对于 DRAM 芯片的读和写两种状态采用不同的仿真模板来完成。控制芯片到 DRAM 芯

片的写状态仿真模板如图 11-15 所示，其顶层结构和 CA 的仿真类似，用户也可以“Push Into”每一个单元查看具体设置结构。

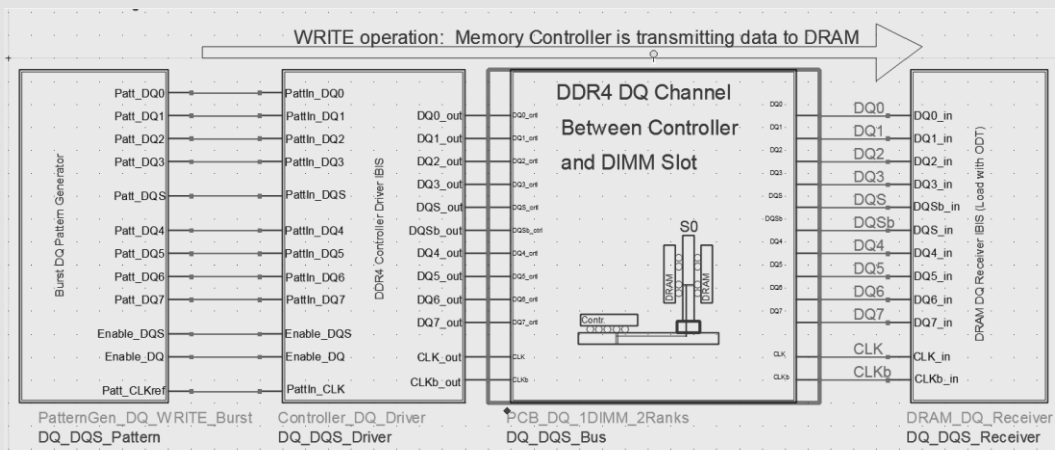


图 11-15 DRAM 写状态仿真模板

(2) 与 CA 总线的仿真类似，对于 PCB 通道的模型也提供两种典型拓扑结构，分别是“PCB\_DQ\_1DIMM\_2Ranks”和“PCB\_DQ\_Sparam”模型，如图 11-16 所示。其中前者是由 ADS 中的 Multi-layer 传输线模型搭建的，而后者则用测量或者仿真得到的 24 端口 S 参数模型来表征。用户如果有自己的传输线模型或是仿真的 S 参数，可以通过 Push Into Hierarchy 对这两个例子的底层设计进行修改。

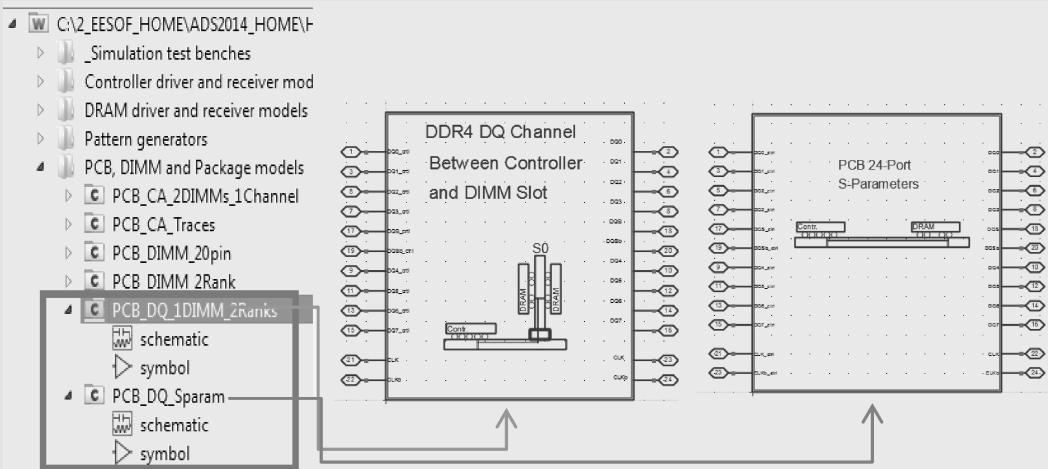


图 11-16 PCB 通道模型

(3) 对于仿真的信号源码型，写周期仿真做了如下设定：

- ⊙ DQ0 ~ DQ7 采用伪随机码型，其数据速率与 SpeedGrade 参数设置一致。
- ⊙ CLK 信号采用 0101 码型，其数据速率为 DQ 的 1/2。
- ⊙ DQS 信号采用 0101 码型，其数据速率与 DQ 相同，同时 DQS 信号被延时 1/2 UI 以确保其边沿与 DQ 信号的中心对齐。



(4) 由于对于 DQ 和 DQS 数据主要关注其时序关系, 因此仿真结束后可以在数据显示窗口中观察其波形, 如图 11-17 和图 11-18 所示。可以看到 DQS 数据的边沿位于 DQ 数据的中心。对于时序关系的测量, 由于项目很多, 我们会在后续的一致性测量中自动进行。

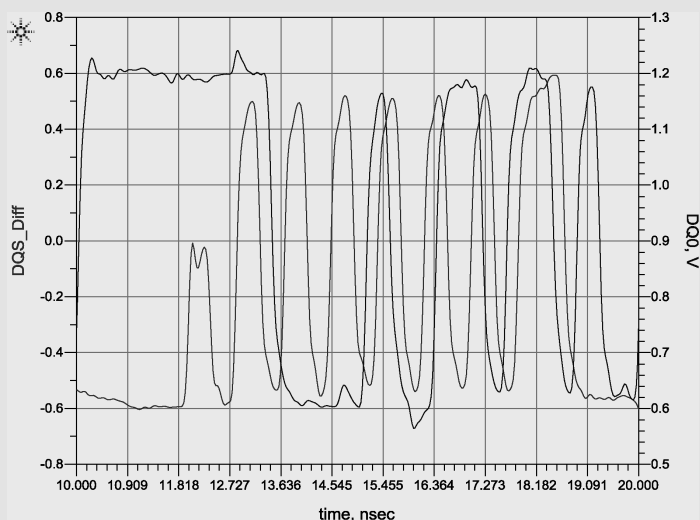


图 11-17 DQ 和 DQS 波形

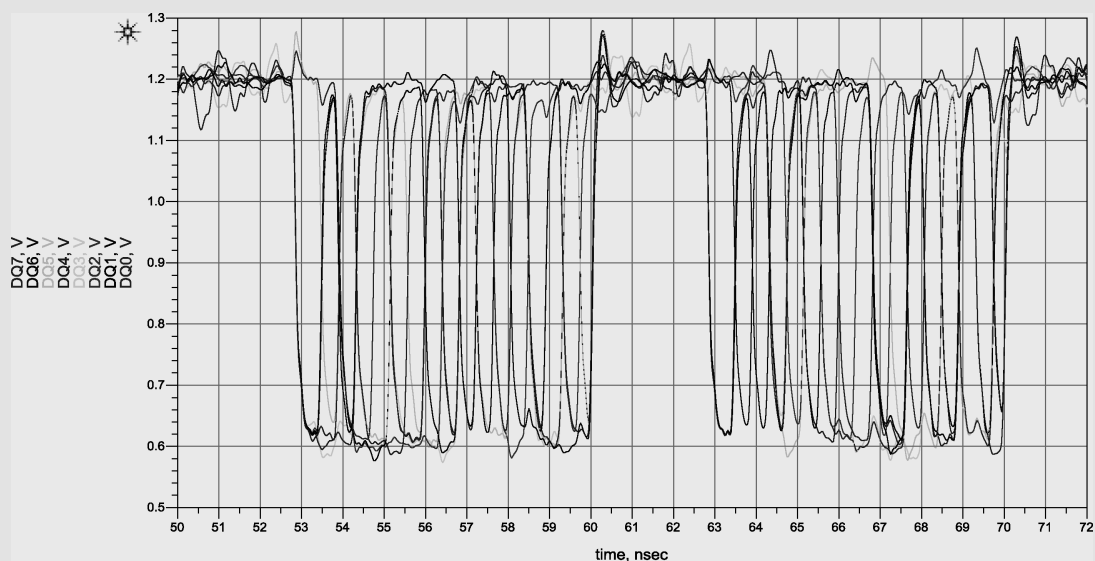


图 11-18 DQ 信号波形

(5) 最后的波形数据也会被存于/data/waveform/DDR4\_Write 文件夹中, 如图 11-19 所示。

## 6. DQ\_READ 仿真

(1) 对于读周期的仿真来讲, 最大的不同在于 DRAM 芯片变成发射端, 而控制芯片成为接收端, 如图 11-20 所示, 因此两边的 IBIS 模型需要完成 Input/Output 类型的互换, 如图 11-21 所示, 所以 PetternGen 单元从左边移到了右边, 作为 DRAM 端的激励。

Name	Date modified	Type	Size
<input type="checkbox"/> CLK.h5	12/25/2014 2:58 PM	H5 File	23 KB
<input type="checkbox"/> CLK_Diff.h5	12/25/2014 2:58 PM	H5 File	23 KB
<input type="checkbox"/> CLKb.h5	12/25/2014 2:58 PM	H5 File	23 KB
<input type="checkbox"/> DQ0.h5	12/25/2014 2:58 PM	H5 File	23 KB
<input type="checkbox"/> DQ1.h5	12/25/2014 2:58 PM	H5 File	23 KB
<input type="checkbox"/> DQ2.h5	12/25/2014 2:58 PM	H5 File	23 KB
<input type="checkbox"/> DQ3.h5	12/25/2014 2:58 PM	H5 File	23 KB
<input type="checkbox"/> DQ4.h5	12/25/2014 2:58 PM	H5 File	23 KB
<input type="checkbox"/> DQ5.h5	12/25/2014 2:58 PM	H5 File	23 KB
<input type="checkbox"/> DQ6.h5	12/25/2014 2:58 PM	H5 File	23 KB
<input type="checkbox"/> DQ7.h5	12/25/2014 2:58 PM	H5 File	23 KB
<input type="checkbox"/> DQ5.h5	12/25/2014 2:58 PM	H5 File	23 KB
<input type="checkbox"/> DQS_Diff.h5	12/25/2014 2:58 PM	H5 File	23 KB
<input type="checkbox"/> DQ5b.h5	12/25/2014 2:58 PM	H5 File	23 KB

图 11-19 仿真波形保存为 .h5 文件

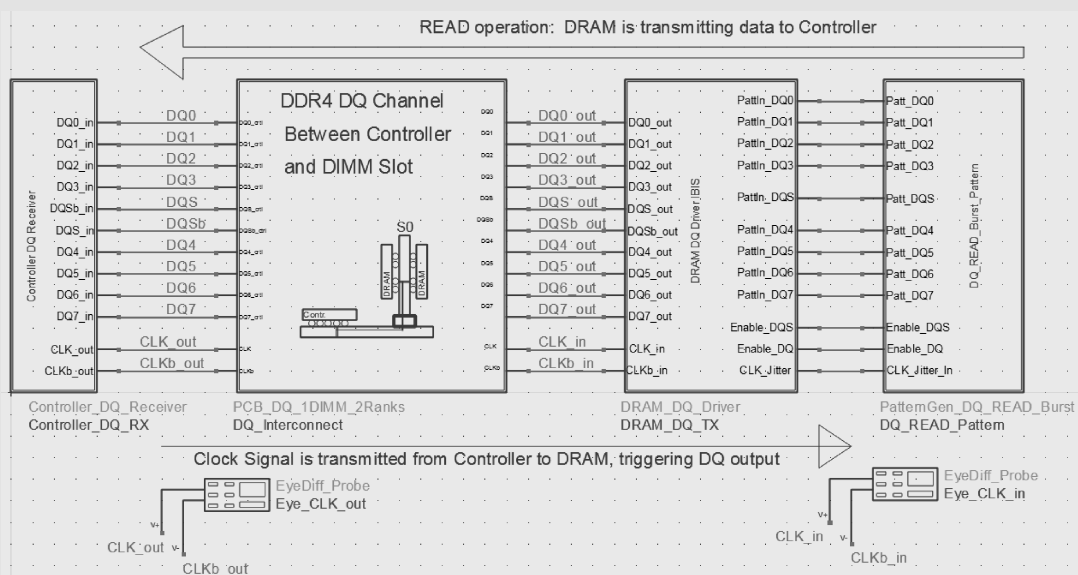


图 11-20 DRAM 读状态仿真

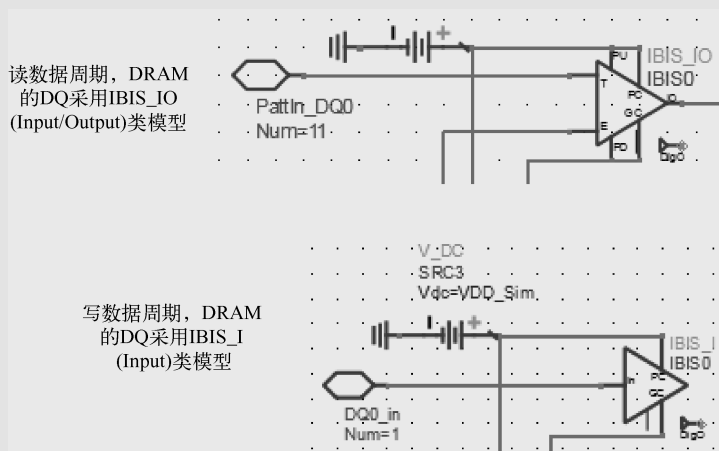


图 11-21 读状态和写状态时 IBIS 模型的选择

(2) 另外一个不同在于 DQ 和 DQS 数据在写周期中需要边沿对齐, 因此 DQS 激励信号无须 0.5UI 的延时, 如图 11-22 所示。

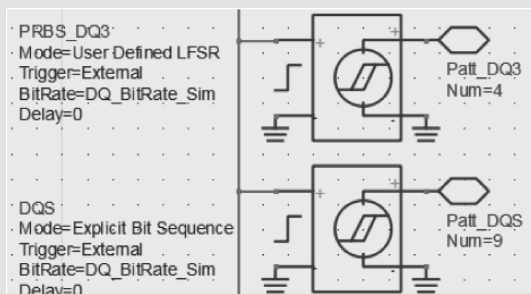


图 11-22 写状态时的 DQ 与 DQS 信号

(3) 最后仿真结果如图 11-23 和图 11-24 所示。

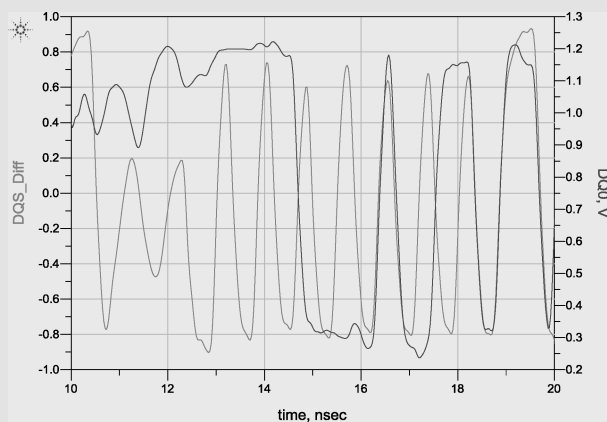


图 11-23 DQ 和 DQS 波形

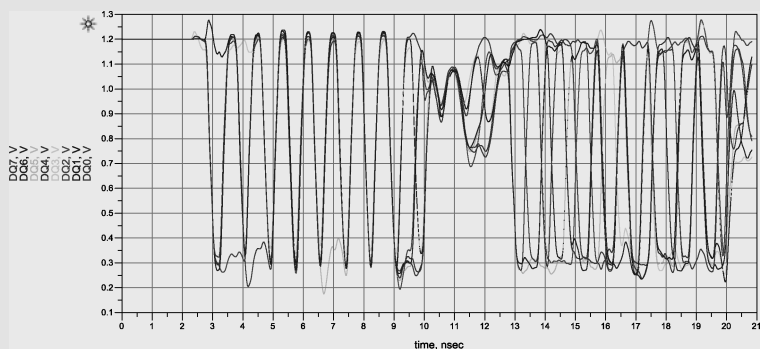


图 11-24 DQ 信号波形



## 11.1 DDR4 一致性测试关键指标

JEDEC 针对 DDR4 的一致性测试规定了很多指标, 关键指标包括建立/保持时间 (Set-

up/Hold time)、飞行时间 (Flight Time)、信号斜率 (Skew Rate)、过冲/下冲 (Overshoot/Undershoot) 等。

同时，JEDEC 对于每种参数也规定了测量算法。例如，图 11-25 ~ 图 11-27 分别定义了单端信号斜率、过冲/下冲面积以及差分信号电压交叉点  $V_{ix}$  的算法 (资料引用自 JEDEC: JEDEC79 -4 DDR4 SDRAM Specification)。有关其他 DDR4 测试标准，大家可以参看如下链接中 JEDEC 相关的规范，在此不予一一赘述。

<http://www.jedec.org/standards - documents/docs/jesd79 -4>

Table 76 — Single-ended output slew rate definition

Description	Measured		Defined by
	From	To	
Single ended output slew rate for rising edge	$V_{OL}(AC)$	$V_{OH}(AC)$	$[V_{OH}(AC)-V_{OL}(AC)] / \Delta TRse$
Single ended output slew rate for falling edge	$V_{OH}(AC)$	$V_{OL}(AC)$	$[V_{OH}(AC)-V_{OL}(AC)] / \Delta TFse$

NOTE :  
1. Output slew rate is verified by design and characterization, and may not be subject to production test.

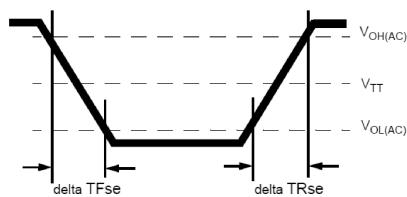


图 11-25 单端信号斜率的算法

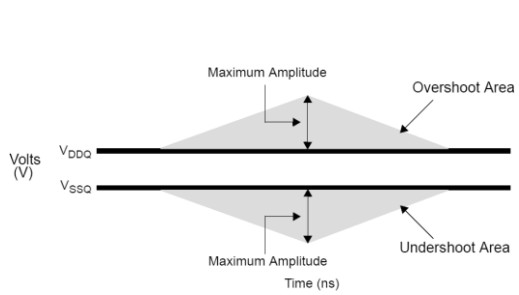


图 11-26 Overshoot/Undershoot 面积算法

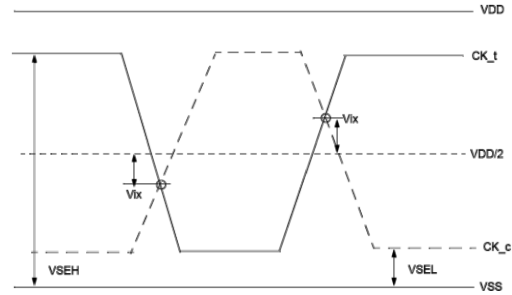


图 11-27 差分信号线交叉点  $V_{ix}$  算法

由此可以看出，DDR4 一致性测试所涵盖的参数众多，如果单靠手工测量，非常耗时耗力。因此 Keysight 公司推出了全自动的基于 Infiniium 的 DDR3、DDR4 等一次性测试软件，只需简单的几步设置，即可自动高效地完成全部一致性测试项目并自动生成测试报告。

同时 ADS DDR4 Test Bench 所支持的 Waveform Bridge 功能，可将所有仿真波形自动导出成 Infiniium 所支持的格式，方便用户直接在 Infiniium 中进行一致性测试，最大限度地保证了仿真和实测的一致性。



## 11.2 DDR4 一致性测试步骤

在仿真流程中我们仿真了不同的读周期、写周期和 CA 的波形，在此节中以 DDR4\_

Write 波形为例看看如何在 Infiniium 中采用 DDR4 分析软件完成写周期的一致性测试。

(1) 打开 Infiniium 脱机版软件。

(2) 从 Analyze→Automated Test Apps→N6462A/N6462B DDR4 Test App 打开一致性测试分析界面, 如图 11-28 所示。

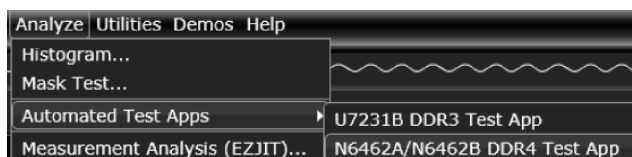


图 11-28 Infiniium 软件中的一致性测试选项

(3) 在分析界面中, Speed Grade 选中 DDR\_2400, 与仿真数据速率一致, 然后单击“Offline Setup”按钮, 如图 11-29 所示。

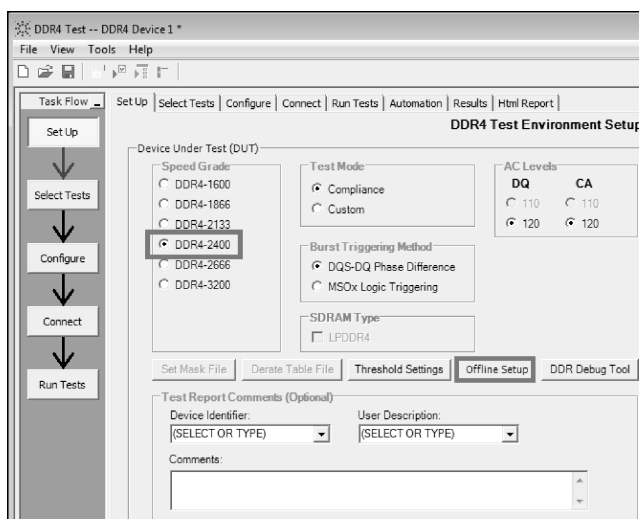


图 11-29 DDR4 一致性测试界面

(4) 如图 11-30 所示, 在新弹出的 Offline Processing 界面中, 先选中“Enable Offline Processing”, 用“Browse”按钮选择相应波形, 然后单击“Done”按钮关闭窗口。注意在写周期仿真中没有包含 CS、CA 信号, 因此写周期的一致性测试主要检验时序关系以及 DQ、DQS 波形等。

波形文件的对应关系如下:

☺ Clock:	CLK_Diff. h5
☺ DQS Differential:	DQS_Diff. h5
☺ Data (DQ) /Data Mask (DM):	DQ0. h5
☺ DQS Plus:	DQS. h5
☺ DQS Minus:	DQSb. h5
☺ CLK Plus:	CLK. h5
☺ CLK Minus:	CLKb. h5

(5) 如图 11-31 所示, 进入“Select Tests”界面, 选中相关测试项, 由于我们在上一步没有提供 CA、CS 相关波形, 因此并不是所有测试项都可选, 在测试列表中所有标记为绿

色圆圈的都是可选项目。

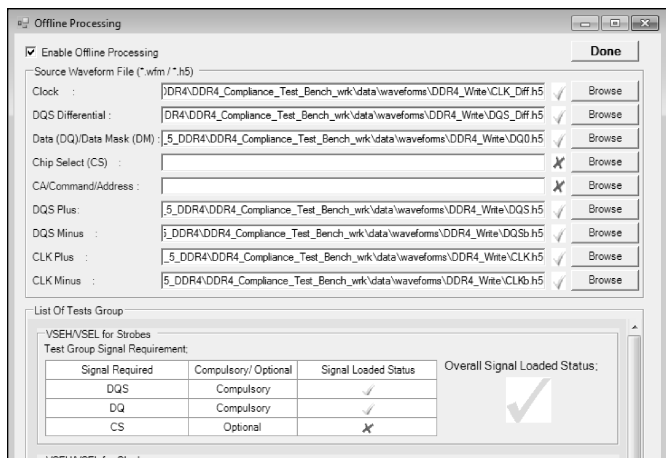


图 11-30 选择离线波形文件

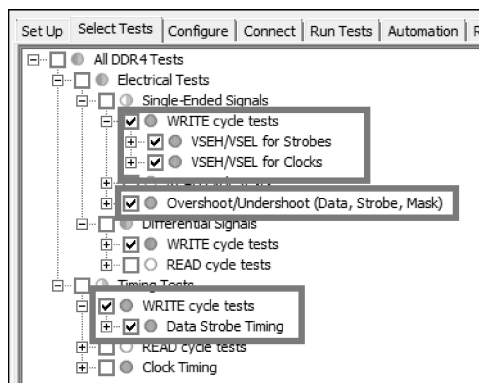


图 11-31 选择测试项

(6) “Configure” 设置页面中主要是一些和标准相关的参数，如 VDD、VTT 等，如图 11-32 所示。用户可以根据自己的芯片特性自行选择，点中每一个参数，在右边都会有相应的说明文档以及可选的数值下拉列表，可以方便用户了解参数的具体含义。在我们的例子中这里可以不做修改。

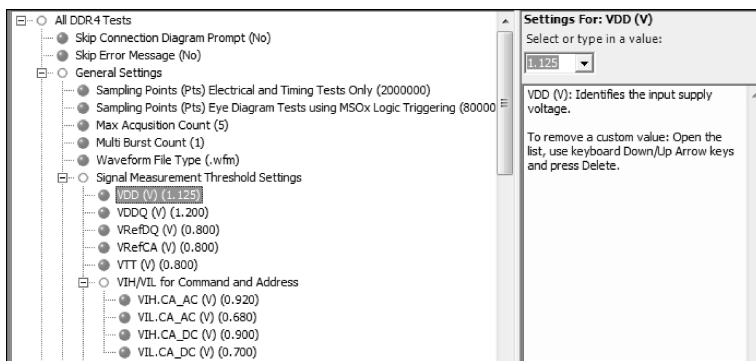


图 11-32 “Configure” 设置页面

(7) 随后在“Connect”设置页面中会提示一些实测的连接方式，对于仿真波形的测试，只需直接勾选“I have completed these instructions”并单击“Run Tests”按钮启动测试，如图 11-33 所示。

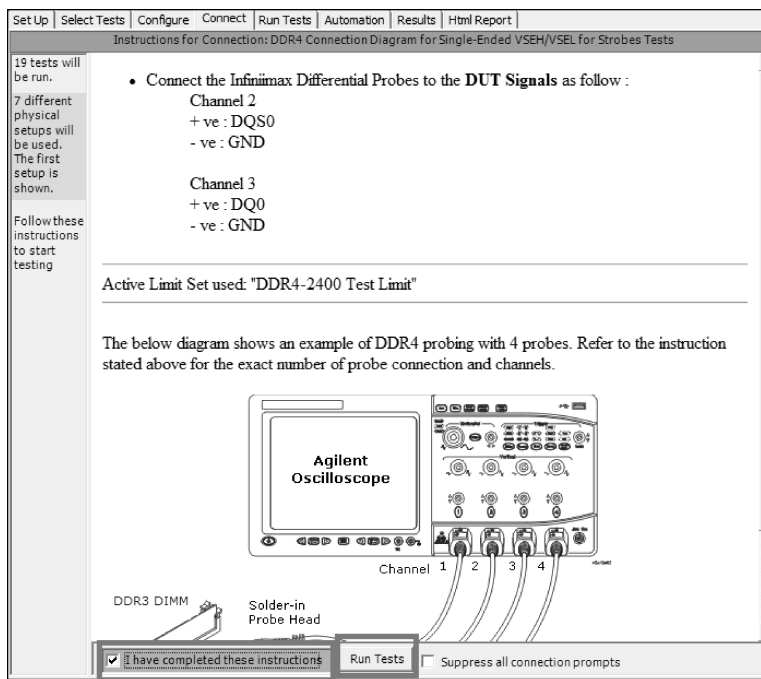


图 11-33 “Connect”设置页面

(8) 在测试过程中用户可以在 Infiniium 界面观测实际波形测试过程，测试结束后会自动生成测试报告，用户可以单击“Html Report”标签页来查看，如图 11-34 所示。例如，在我们的例子中，总共测试了 38 项，37 项结果为通过，而有 1 项显示失败。

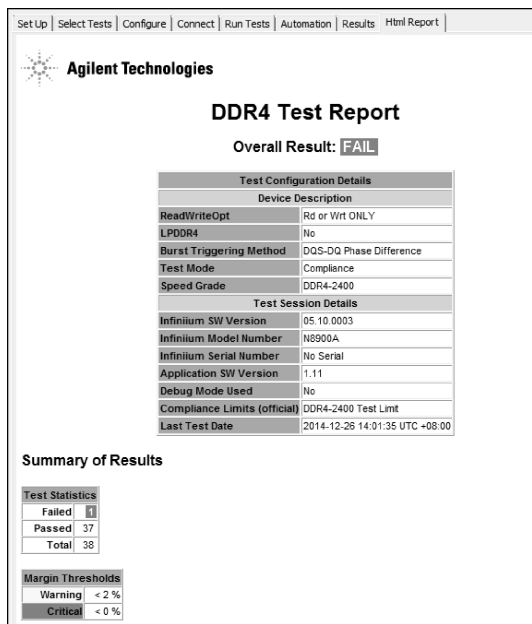


图 11-34 测试报告生成

(9) 针对每一项 JEDEC 规范中的定义, 报告中能自动显示当前实测值以及基于标准算出的设计裕量等, 如图 11-35 所示。

Set Up	Select Tests	Configure	Connect	Run Tests	Automation	Results	Html Report	
①	3	WVST						Information Only
✓	0	1	th(CC) Rising Edge Measurements			28 ps	66.3 %	VALUE <= 83 ps
①	1	1	th(Cavg) Rising Edge Measurements					Information Only
✓	0	1	th(Per) Rising Edge Measurements			-18 ps	28.6 %	-42 ps <= VALUE <= 42 ps
①	1	1	thri(2er) Rising Edge Measurements					Information Only
①	1	1	thri(3er) Rising Edge Measurements					Information Only
①	1	1	thri(4er) Rising Edge Measurements					Information Only
①	1	1	thri(5er) Rising Edge Measurements					Information Only
①	1	1	thri(6er) Rising Edge Measurements					Information Only
①	1	1	thri(7er) Rising Edge Measurements					Information Only
①	1	1	thri(8er) Rising Edge Measurements					Information Only
①	1	1	thri(9er) Rising Edge Measurements					Information Only
①	1	1	thri(10er) Rising Edge Measurements					Information Only
①	1	1	thri(11er) Rising Edge Measurements					Information Only
①	1	1	thri(12er) Rising Edge Measurements					Information Only
①	1	1	thri(ner) Rising Edge Measurements					Information Only
✓	0	1	th(Average High Measurements			501.256170166 mCK(avg)	46.9 %	480.000000000 mCK(avg) <= VALUE <= 520.000000000 mCK(avg)
✓	0	1	th(L Average Low Measurements			498.743829834 mCK(avg)	46.9 %	480.000000000 mCK(avg) <= VALUE <= 520.000000000 mCK(avg)
①	1	1	th(duty-high) Jitter Average High Measurements					Information Only
①	1	1	th(duty-low) Jitter Average Low Measurements					Information Only
①	2	2	VSEH(Strobe)					Information Only
①	2	2	VSEL(Strobe)					Information Only
①	2	2	VSEH(Clock)					Information Only
①	2	2	VSEL(Clock)					Information Only
✓	0	1	Overshoot amplitude (Data, Strobe, Mask)			46.720000000 mV	88.3 %	VALUE <= 490.000000000 mV
✓	0	1	Overshoot area (Data, Strobe, Mask)			6.402305000 mV-ns	96.8 %	VALUE <= 200.000000000 mV-ns
✓	0	1	Undershoot amplitude (Data, Strobe, Mask)			-560.560000000 mV	275.2 %	VALUE >= 320.000000000 mV
✓	0	1	Undershoot area (Data, Strobe, Mask)			0.000000000000 V-ns	100.0 %	VALUE <= 100.000000000 mV-ns
✓	0	1	Vshdiff_CK(A/C)			1.158400000000 V	382.7 %	VALUE >= 2*(VHAC_CA_Volt-VrefCA_Volt) V
①	1	1	VHdiff_CK(D/C)					Information Only
✓	0	1	VLdiff_CK(A/C)			-1.143530000000 V	376.5 %	VALUE <= 2*(VLAC_CA_Volt-VrefCA_Volt) V
①	1	1	VLdiff_CK(D/C)					Information Only
✓	0	1	VHdiff_DQS(A/C)			498.030000000 mV	107.5 %	VALUE >= 2*(VHAC_DQ_Volt-VrefDQ_Volt) V
①	1	1	VHdiff_DQS(D/C)					Information Only
✓	0	1	VLdiff_DQS(A/C)			-513.270000000 mV	113.9 %	VALUE <= 2*(VLAC_DQ_Volt-VrefDQ_Volt) V
①	1	1	VLdiff_DQS(D/C)					Information Only
✗	1	1	VW(CK)			325.617000000 mV	-85.7 %	-120.000000000 mV <= VALUE <= 120.000000000 mV

图 11-35 测试报告的内容

(10) 而列表中最后一项 VIX (CK) 显示没有通过, 原因是超出标准的限定范围 ( -120 ~ 120mV), 只需双击这个测试项, 就可查看具体的测试波形图, 如图 11-36 所示。

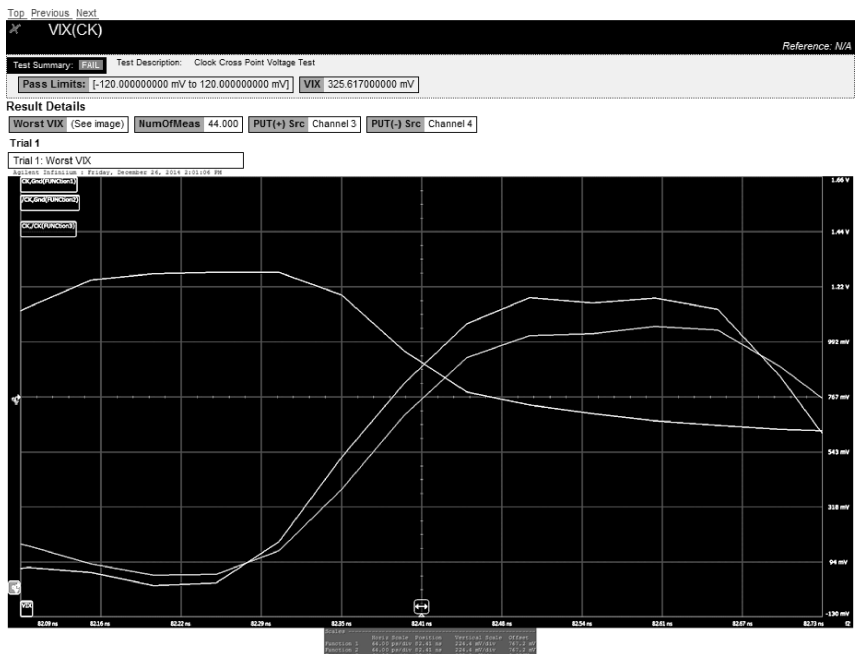


图 11-36 未通过测试项的波形



**【总结】**

本实例介绍了使用 ADS 软件的 DDR4 Test Bench 模板快速搭建 DDR4 拓扑结构进行仿真，并通过模板中提供的 Waveform Bridge 功能将仿真波形自动导出成 Keysight 示波器分析软件 Infiniium 所支持的格式，并在 Infiniium 中进行一致性测试的流程。采用这样的从仿真到测量的一体化解决方案，可以大大缩短用户的设计周期，从以前的几周时间缩短到几天之内完成。

# 实例 12    DDR BUS 仿真器

【目的】

与现今主流的 DDR3 相比，新一代的 DDR4 一致性测试标准除了保留了原有的时序测量项目以外，由于 DDR4 更高的数据速率导致其时间周期大大缩短，从而随机抖动对于误码率的影响变得不可忽略，因此在 JEDEC 最新的 DDR4 测试标准中加入了 DQ 眼图模板（Eye Mask）测试。针对这一新的需求，ADS 在 2015 版本推出了最新的 DDR BUS（并行链路）仿真器，相比较于传统的 Transient 仿真器，可以高效地仿真低至  $1\text{E} - 16$  BER（误码率）水平的眼图，从而使得 DDR4 眼图模板测量成为可能。

本实例会系统介绍如何使用 DDR BUS 仿真器来进行 DDR4 眼图仿真及眼图裕量的测试。



## 12.1    DDR4 一致性测试面临的挑战

与 DDR3 相比，DDR4 的典型变化包括数据速率从 2133Mb/s 提高到 3200Mb/s（见图 12-1），这使得每个 bit（比特）的时间周期显著减小；另外 DDR4 采用了更低的 1.2V 电压，这也使得其眼图的最大纹波（Ripple）裕量从 75mV 降为 60mV。因此对于 DDR4 的应用来说，随机抖动（RJ）和码间干扰（ISI）对于眼图的影响变得无法忽略。

基于以上特性，JEDEC 新的 DDR4 标准中加入了 DQ Eye Mask 的测试，并规定测试眼图需要能够反映低至  $1\text{E} - 16$  误码率量级，以此更准确地预估 Rj 和 ISI 在极低误码率水平下对于输出眼图的影响。JEDEC 中 DDR4 的 DQ 眼图模板分别由 Deterministic Mask 和 Random Mask 两部分组成（见图 12-2），Random 的模板反映了随机抖动对于眼图的影响。

	DDR3	DDR4
Speed	2133 Mb/s	3200 Mb/s (goal)
3 <sup>rd</sup> Harmonic	3.2 GHz	4.8 GHz
Unit Interval	469 pS	312 pS
Vdd	1.5 V	1.2 V
Max Ripple	+/- 75 mV	+/- 60 mV

图 12-1    DDR3 和 DDR4 标准的主要区别

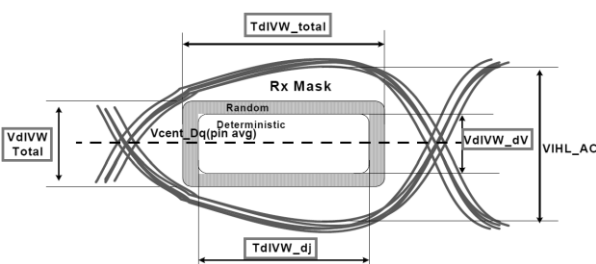


图 12-2    DDR4 标准中对眼图模板的定义  
(JEDEC Standard No. 79 - 4 Page 198)

图 12-3 中给出了 JEDEC DDR4 眼图模板的参数定义。其中眼图裕量基于外围的 Random 模板来测量，其所对应的 BER level 为  $1\text{E} - 16$ 。而对于 Deterministic mask，标准中则规定其

值由芯片设计者来决定，并且测量标准暂时待定。

Table 103 — DRAM DQs In Receive Mode; \* UI=tck(avg)/min/2

Symbol	Parameter	1600,1866,2133		2400		2666,3200		Unit
		min	max	min	max	min	max	
VdIVW_total	Rx Mask voltage - p-p total	-	136 (note12)	-	tbd	-	tbd	mV
VdIVW_dV	Rx Mask voltage - deterministic	-	136	-	tbd	-	tbd	mV
TdIVW_total	Rx timing window total	-	0.2 (note12)	-	tbd	-	tbd	UI*
TdIVW_dj	Rx deterministic timing	-	0.2	-	tbd	-	tbd	UI*

图 12-3 JEDEC DDR4 眼图模板的参数定义 (JEDEC Standard No. 79 - 4 Page 202)

为了更直观地反映随机抖动在高数据速率下对于眼图的影响，图 12-4 给出了一个通道眼图仿真实例，相同的通道模型，在 800Mb/s 速率下随机抖动 (RJ) 和码间互扰 (ISI) 对于眼宽和眼高的影响非常小，在  $1E-3$  和  $1E-16$  BER level 下眼图的等位线 (Contour) 变化很小，随机抖动对于估算设计裕量的影响不大，因此仿真通常只需要跑几千 bit 即可确认设计裕量，因此常规的 Transient 仿真器即可满足仿真需求。

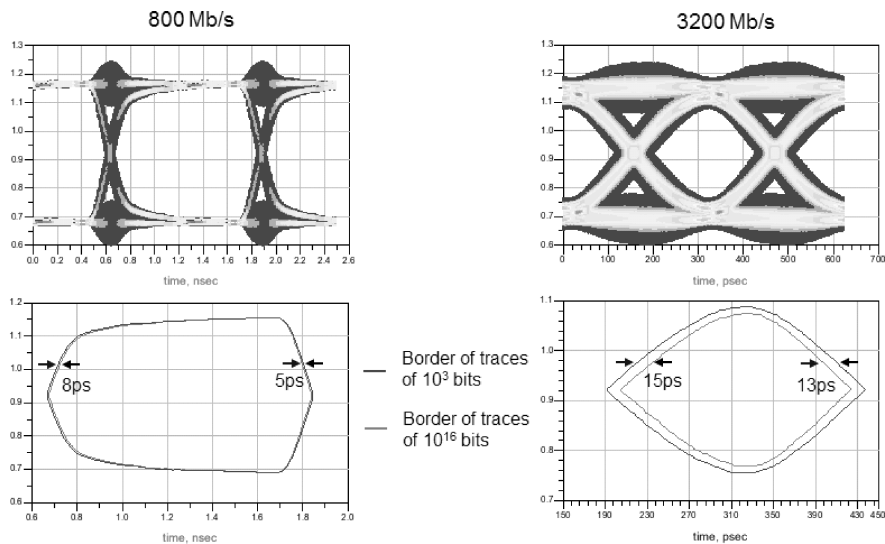


图 12-4 不同速率下的 DDR 眼图仿真实例

然而当数据速率达到 3.2Gb/s 时，由于 UI 大大缩小，使得  $R_j$  和 ISI 的影响变得不可忽略，可以直观地看到等位线随着 BER level 的变化更为显著，因此 JEDEC 的测试标准中规定了 DDR4 的眼图需要在低至  $1E-16$  BER level 进行眼图模板测试，从而更准确地估计设计裕量，这就要求仿真器能够在短时间内更精准地仿真出低至  $1E-16$  BER 量级的眼图，采用传统的 Transient 仿真器往往无法实现。

基于此种需求 ADS 中 DDR BUS Simulator 应运而生，采用创新的统计学算法可以在短时间内计算出极低误码率量级的眼图并自动根据眼图模板测试裕量。眼图裕量测量方法如图 12-5 所示。

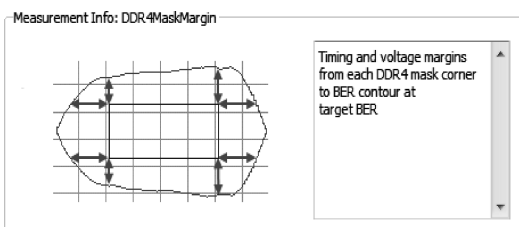


图 12-5 眼图裕量测量方法



## 12.2 DDR BUS 仿真设置实例

与传统的 Transient 仿真器对比, DDR BUS Simulator 具有如下特性:

- ☺ 基于通道的冲击响应, 采用创新的统计学算法, 可以快速计算出低至  $1\text{E}-16$  误码率下的 DQ 和 DQS 信号的眼图特性。
- ☺ 基于用户定义的眼图模板可自动进行眼图裕量的测量。
- ☺ 对于 Tx/Rx 的模型, ADS 支持包括内嵌的行为级模型、IBIS 模型、Spice 模型和 Verilog-A 模型的灵活组合。
- ☺ 支持仿真通道间串扰。
- ☺ 对于内嵌的 Tx 行为级模型, 支持 de-emphasis 和非对称的上升、下降沿设置, 以更好地反映实际芯片的工作状态。
- ☺ 对于内嵌的 Rx 行为级模型, 支持 CLTE (连续时域线性均衡) 算法的设置。

下面通过 ADS 自带的 DDR BUS 仿真实例, 详细介绍 DDR BUS Simulator 的仿真设置。

(1) 从 ADS 主窗口选择 Open→Example, 打开 ADS2015\example\HSD 文件夹下面的 DDRBusSimulator\_wrk.7zads 工程文件, 如图 12-6 所示。

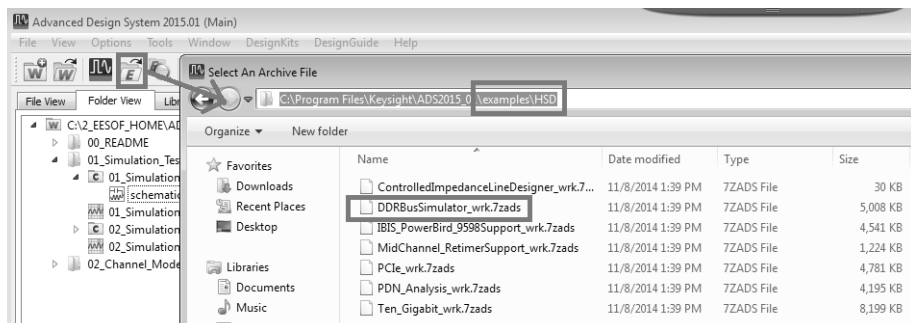


图 12-6 打开 DDR4 仿真实例

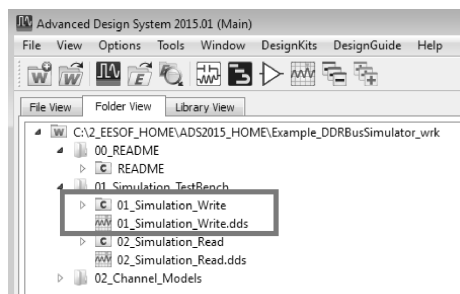



图 12-7 打开 01\_Simulation\_Write 原理图

(2) 此工程文件中包含两个主要的仿真模板, 分别为读状态和写状态, 两种模板的设置基本相同, 只是 Tx/Rx 模型相互交换。在此案例中我们通过写状态的模板来介绍 DDR BUS 的仿真设置。打开 01\_Simulation\_Write 这个设计单元中的原理图, 如图 12-7 所示。

(3) 可以看到仿真模板中包含了 Tx/Rx 的行为级模块以及 Controller board 模型、DIMM Socket 模型和 DIMM Board 模型, 针对这些模型, 用户都可以替换为自己的模型并进行仿真, 如图 12-8 所示。有关通道模型的细节可以通过单击相应的模块, 并通过单击 Push Into  按钮来查看。

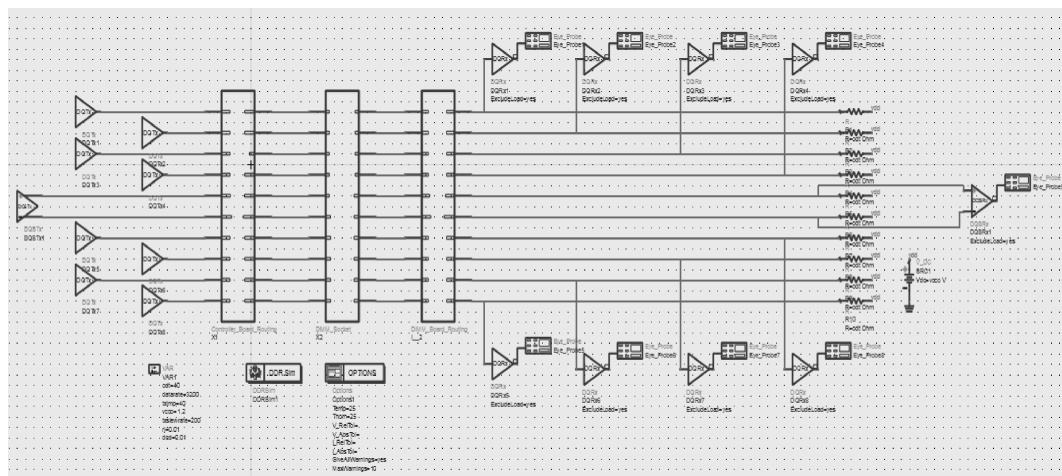


图 12-8 01\_Simulation\_Write 原理图

(4) 在器件 Palette 中可以发现一个新的类别: Simulation - DDR, 大部分和 DDR Simulation 相关的 Tx/Rx 模型以及仿真控件都可以在这里找到。双击模板中的任何一个 DQTx 或者 DQSx 元件, 可以看到如图 12-9 所示的设置。

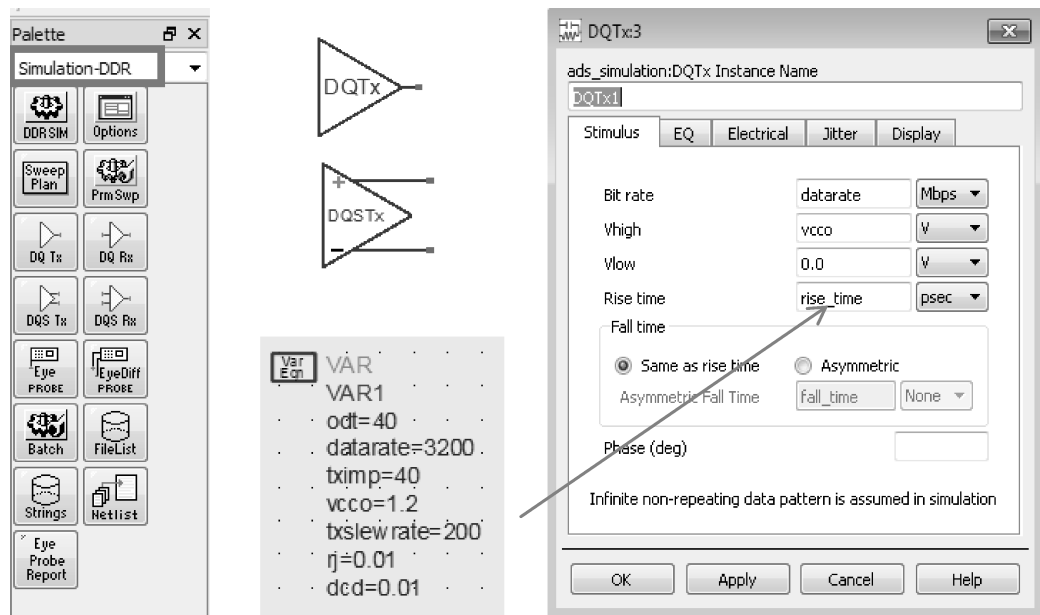


图 12-9 Simulation - DDR 元件库中的 DQ 和 DQS 元件

(5) 在 DQTx 设置窗口可以分别设置激励源 (Stimulus)、均衡参数 (EQ)、负载 (Electrical)、信号抖动 (Jitter) 以及模块显示项目 (Display), 如图 12-10 所示。因为 DDR BUS 采取统计学的算法, 因此在设置激励源的时候无须指定数据码型 (Pattern), 仿真器会默认采用无重复的随机码型来估算码型对于输出眼图的影响。另外, 因为往往一个并行的 DDR4 拓扑结构中包括多个 Tx 模块, 因此在设置上推荐使用变量定义, 以方便同时修改所有的 Tx 模块设置。

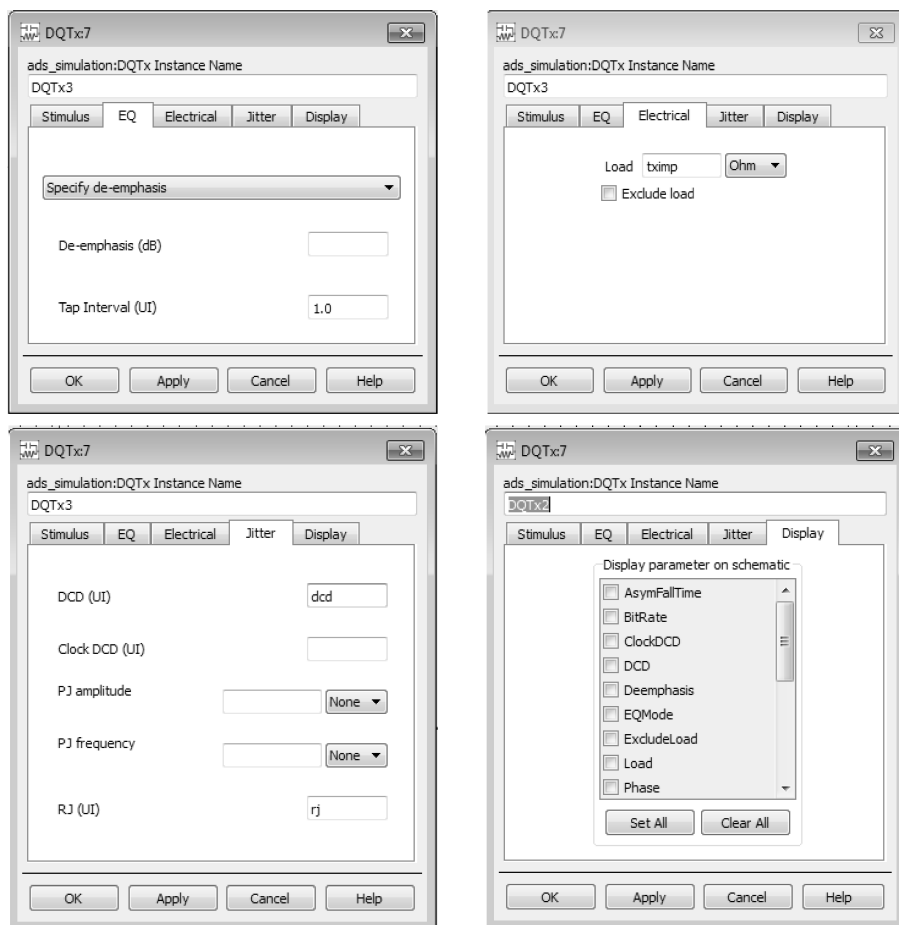


图 12-10 DQTx 元件的参数设置

(6) 双击任何一个 DQRx 或 DQSRx 的模块，可以看到此处用户也可以设置 CTLE 均衡、负载以及 Jitter 的参数，如图 12-11 所示。

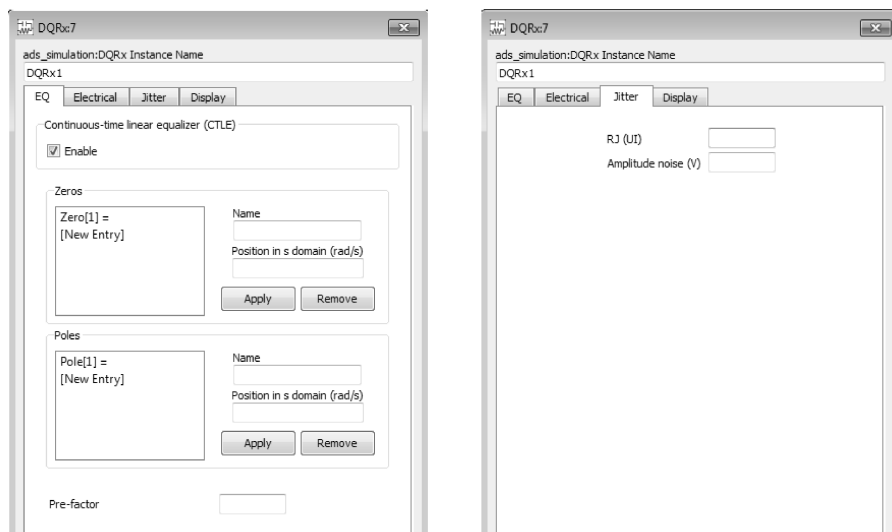


图 12-11 DQRx 元件的参数设置

(7) 另外在 DDR BUS 仿真中数据的测量是通过在通道节点加 Eye\_Probe 来实现的。双击打开任何一个 Eye\_Probe，在“Parameters”设置窗口中主要有两部分需要设置，即 BER Contour 和 DDR4 Mask 数值，如图 12-12 的粗线框所示。其定义如下：

☺ BERContour:

- ◇ BER Value: 是指在测量 Hight@ BER 和 Width@ BER 两个参数时所对应的 BER 值，同时在做眼图裕量测量的时候也是基于这个 BER Value。因此基于 DDR4 测试标准定义，这里推荐使用  $1e-16$ 。
- ◇ BER list 是指如果在仿真后要显示不同 BER level 下的眼图等位线，则可在此处设置 BER list (如 list( $1e-10$ 、 $1e-12$ 、 $1e-16$ ))。

☺ DDR4 Mask: 根据图 12-13 中 DDR4 的模板定义，通过高度和宽度来定义 Mask。

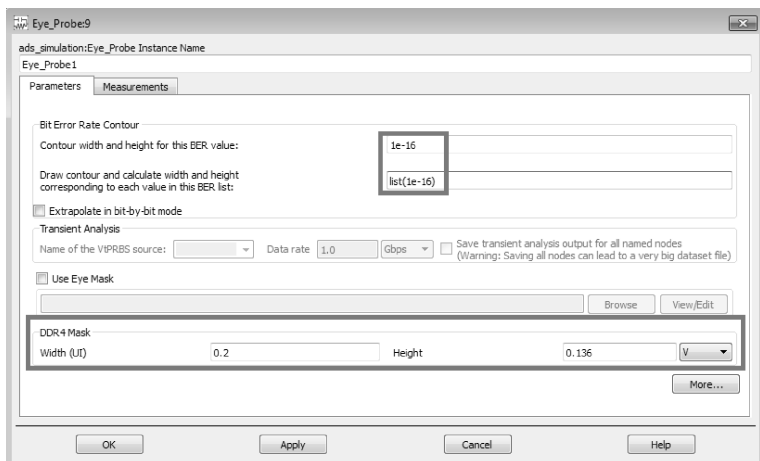


图 12-12 眼图探针的参数设置

(8) 所有仿真后的测量项目都是在 Eye\_Probe 的“Measurements”标签页中设置的，如图 12-13 所示，左侧列出了所有可选项目，如眼图、眼高、眼宽、抖动、等位线等，单击每个项目，在下面的“Measurement Info”框中都会有相应的解释说明。将所需测量的项目通过

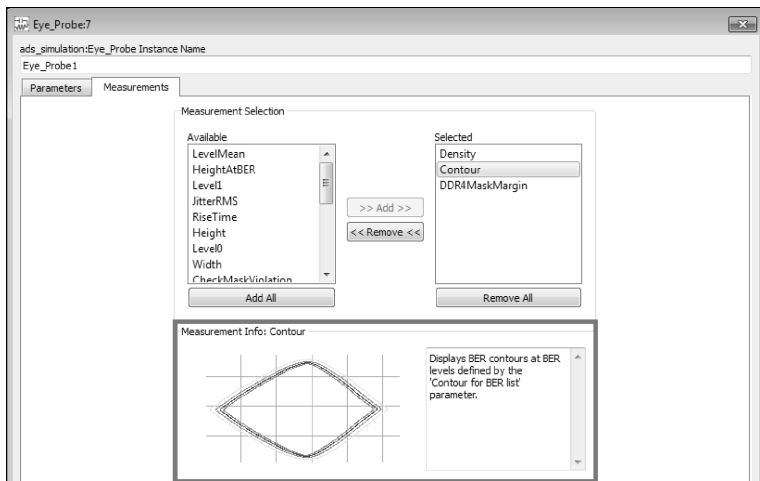


图 12-13 眼图探针的测量项选择

“Add”按钮可以加到右侧列表中。在此例子中我们选择了 Density（眼图）、Contour（等高线）、DDR4MaskMargin（模板裕量测量）这三个选项。

(9) 关于仿真控制器的设置。DDR BUS Simulator 的仿真控件也可以在“Simulation DDR” Palette 中找到。双击打开模板中的 DDRSim 控件设置窗口，在“Analysis”标签页中，如果需要仿真低至  $1\text{E} - 16$  BER 量级的眼图裕量，请勾选“Enable ultra low BER ( $< 1\text{e} - 16$ ) simulation”选项，如图 12-14 所示。另外“Status Level”的数值表示仿真器输出信息的详细程度，如果将其由 2 改为 4，则在仿真状态窗口（Status Window）将会输出更多仿真信息。

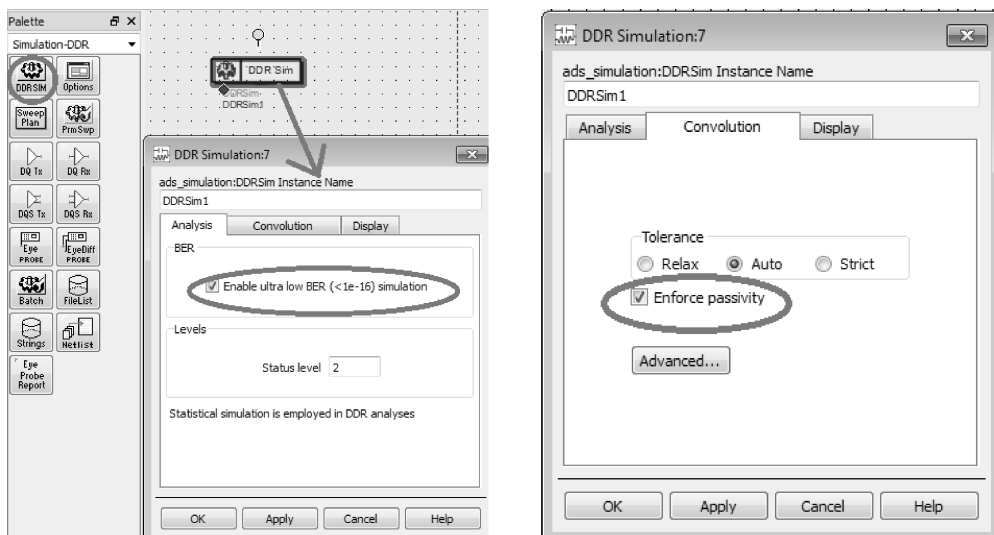



图 12-14 DDR 仿真控制器的设置

在“Convolution”标签页对于大部分仿真可以保持默认设置，不过如果通道中采用的是 S 参数模型呈现有源特性，则建议勾选“Enforce Passivity”选项以自动强制无源。

(10) 最后单击  按钮进行仿真，仿真过后，数据显示窗口（Data Display Window）会自动弹出并显示眼图、 $1\text{E} - 16$  BER level 所对应的等高线以及 DQ 眼图模板裕量测量结果，如图 12-15 所示。

(11) 接下来我们尝试修改均衡参数。将模板中 DQTx1 的 EQ 选项设为“Specify de-emphasis”（自定义去加重），并将 De-emphasis 系数设为 1，如图 12-16 所示。

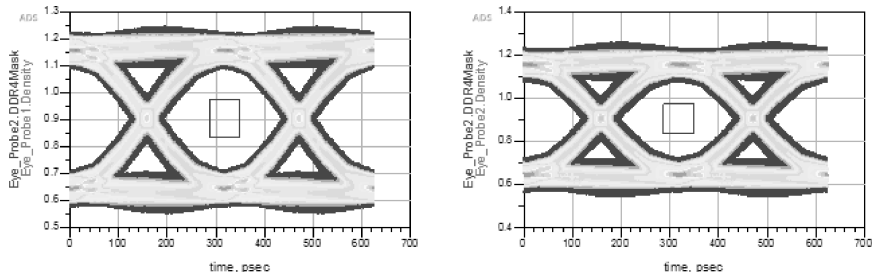


图 12-15 DDR 仿真结果：眼图、误码率等高线及裕量



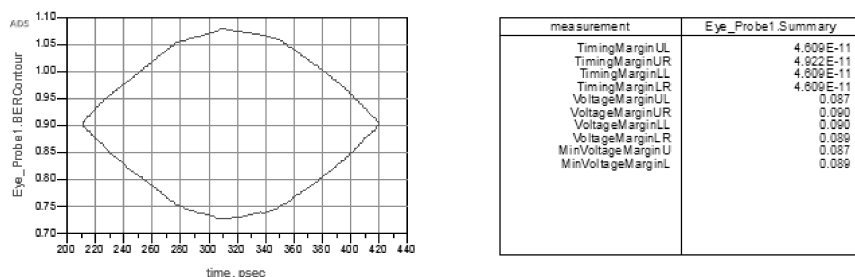


图 12-15 DDR 仿真结果：眼图、误码率等高线及裕量（续）

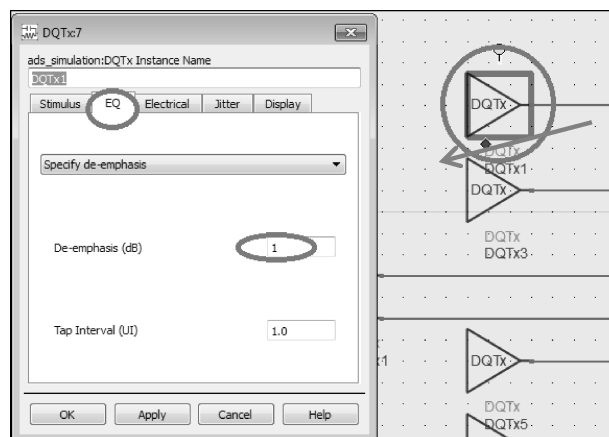


图 12-16 调整均衡参数

(12) 在仿真之前，打开 Simulate→Simulation Settings...，然后将“Use cell name”的选项去掉，并将 Dataset 的名字改为 01\_Simulation\_Write\_new，最后单击“Simulate”按钮，如图 12-17 所示。这样可以将仿真结果另存为 dataset，以免将之前的仿真结果覆盖掉。

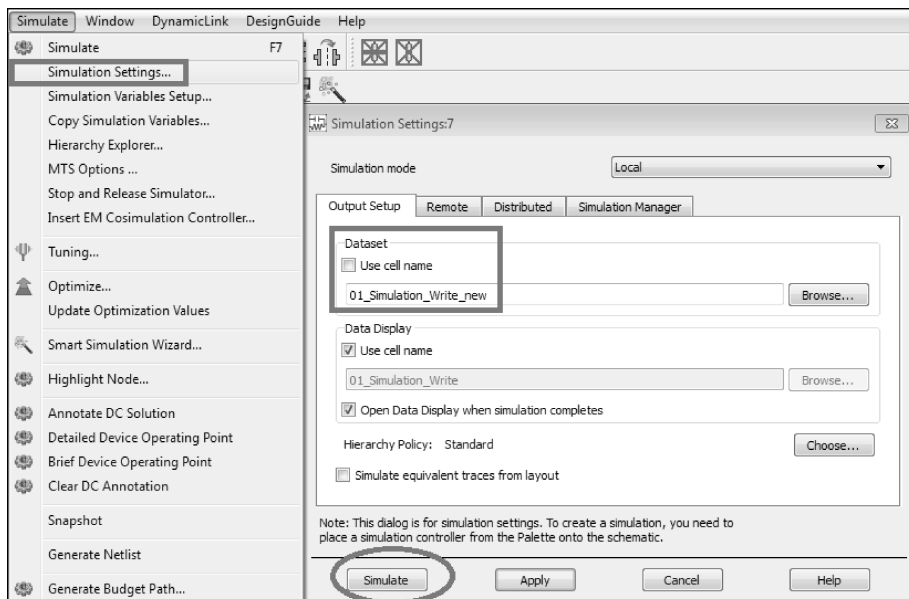


图 12-17 修改仿真结果的保存文件名

(13) 仿真后在弹出的数据显示窗口中, 将 dataset 名字切换到 \_01\_Simulation\_Write\_new, 并对比 Eye\_Probe1 和 Eye\_Probe2 的结果, 如图 12-18 所示。可以看到, 由于 Eye\_Probe1 是加了去加重的结果, 因此眼图和没加去加重的 Eye\_Probe2 结果有所区别, 且其眼图模板裕量的测量结果也普遍好于 Eye\_Probe2 的结果, 说明在 Tx 中加入去加重均衡对输出眼图有改善作用。

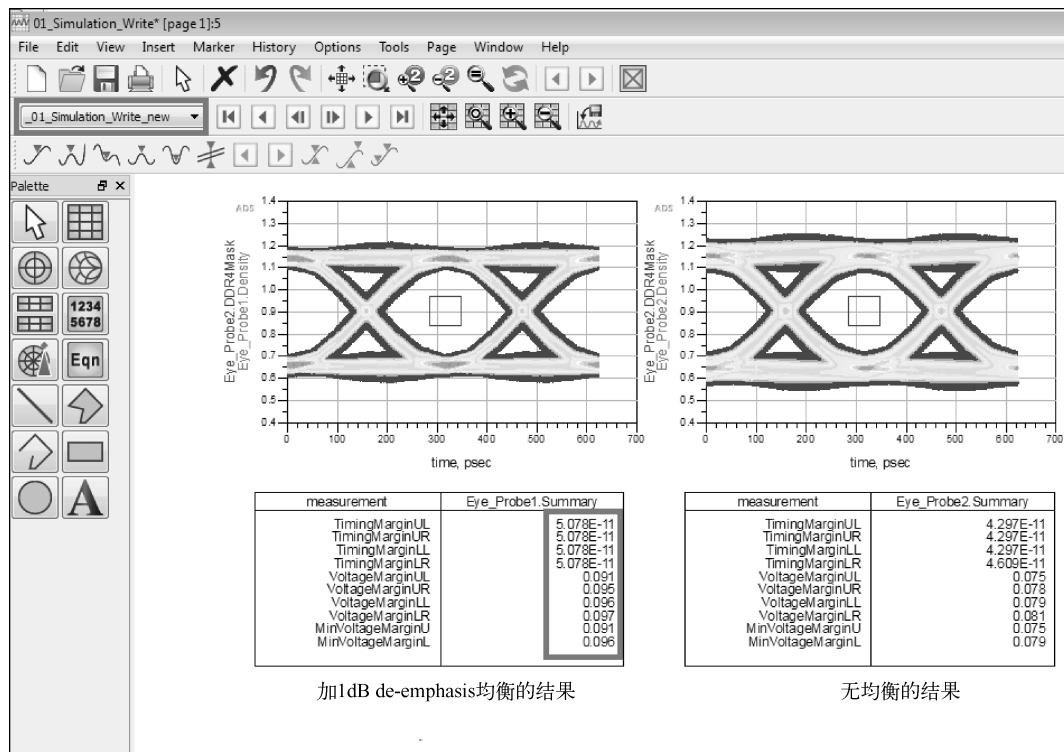


图 12-18 仿真结果比较

### 【总结】

ADS 2015 中最新的并行链路仿真器 (DDR BUS) 专门针对 DDR4 JEDEC 测试标准中所提出的 DQ 眼图模板测量的需求, 采用创新的统计学算法可以快速仿真出极低误码率 ( $BER < 1E-16$ ) 下的眼图等位线 (Contour), 并支持基于眼图模板进行自动的裕量测量。此外, 并行链路仿真器可以支持内嵌的 Tx/Rx 模型、IBIS 模型、Spice 模型等的灵活组合, 最大限度地保障了仿真兼容性, 为用户精准预估  $R_j$  以及 ISI 等对于眼图的影响提供了最有效的仿真手段。

本实例主要通过 ADS 自带的 DDR\_BUS\_Simulator 例子, 介绍了 DDR BUS Simulator 的主要设置、如何通过 Eye-Probe 来自动测量眼图、不同 BER level 下的等高线以及如何自动进行自动的裕量测量等。

# 实例 13 电源完整性分析——电源网络阻抗分析

## 【目的】

本实例通过对 PCB 上电压整流模块（Voltage Regulator Module，VRM）至芯片之间电源网络的阻抗分析，展示如何使用 ADS2015 进行电源完整性分析。

## 【背景知识】

本实例使用的版图为 PandaBoard ES (<http://pandaboard.org/content/resources/references>)。此电路板为 TI OMAP4430 处理器，可安装 Ubuntu 及 Android 等操作系统的微电脑系统。

在电路板中，所有的电源 VRM 皆整合在 Phoenix PMIC 内，其中 1.29V 的网表名称为 VDD\_V1V29，HDMI DAC 电源为 VDD\_VDAC。本实例将对电路板上 1.29V 电源网络的阻抗及 HDMI DAC 电源网络的阻抗进行仿真。



## 13.1 版图的截取与导入

本实例使用 ADS2015 提供的 ADFI 工具将电路板中的需要的部分截取下来，读入 ADS 做进一步的处理及仿真。ADFI 的安装部分可参考实例 10 的内容。以 PandaBoard ES 为例。

(1) 读入 allegro layout file (.brd)：如图 13-1 所示，在 PandaBoard ES 的版图中，左上角 U10 是处理器的部分，PMIC U11 在右下角。在处理器的周围可以看到许多的电容，大多数是去耦电容，在进行版图导出的时候，会将这些电容一起导出送入 ADS 软件中。

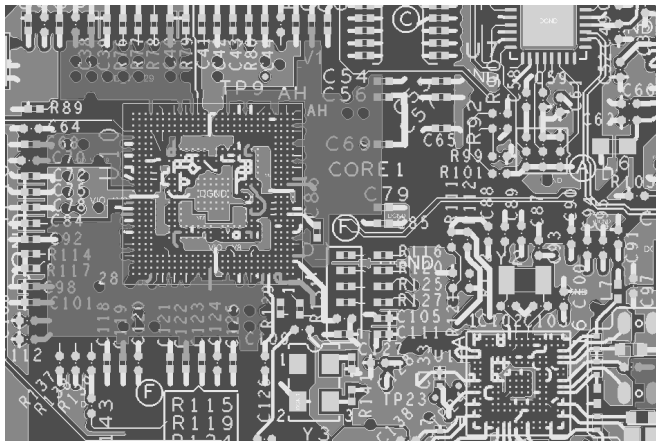


图 13-1 PandaBoard ES 版图设计

(2) 选择需要导出的网表：单击 Allegro 菜单栏 “Export to ADS/EMPro”，选择 “Select Traces” 开启 ADFI 窗口，在窗口左下方激活 “Filter Nets with Wildcards” 功能，并分别在下方输入网表名称 (VDD\_V1V29 及 VDD\_VDAC)，单击 Add-> 将选出的布线网络加入右方 Signal Nets 中。接着输入 DGND 选取 DGND 加入 Ground Nets。完成后如图 13-2 所示。

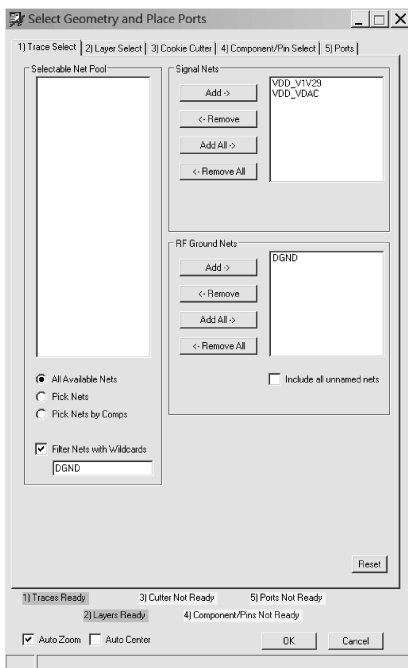


图13-2 Allegro 版图导出——选择网表

(3) 使用 Cookie Cutter 进行版图裁剪：在窗口栏 Cookie Cutter 页面中的 “Expansion distance (MIL):” 中输入 100.00，接着点选 “Build for Signal Nets”，创建裁剪区域，完成后如图 13-3 所示。

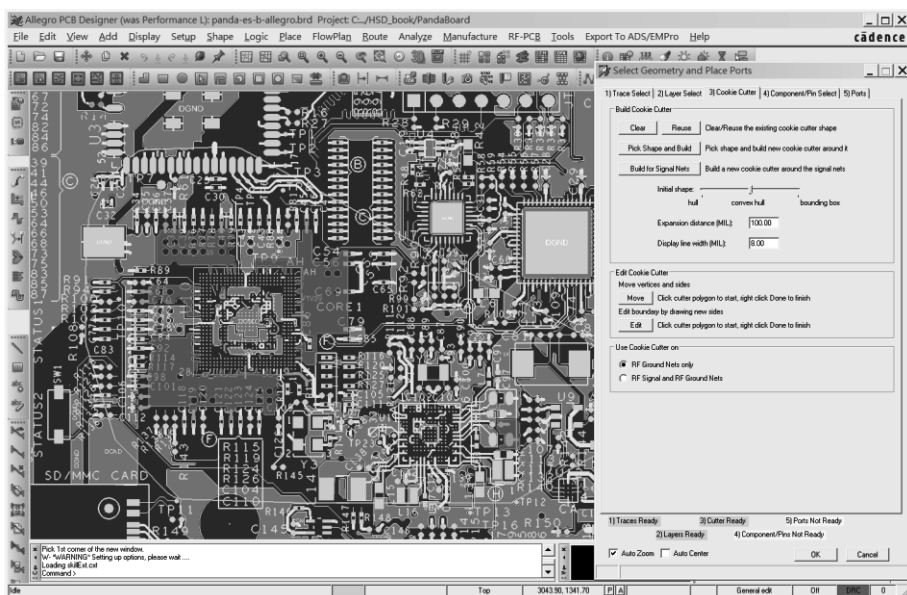


图 13-3 Allegro 版图导出——创建裁剪区域

(4) 选取元件及管脚：切换窗口至“Component/Pin Select”栏。所有连接在已选布线网络上的元件都会自动加入到右上角窗口。在右下角窗口中显示了已经选择的管脚，但所有去耦电容的管脚尚未选入。首先将“Pin Selection”中的“All Selected Nets”切换为“Signal Nets”，选取“Add All”后将右方所有的管脚加入。接着将“Signal Nets”切换回“All Selected Nets”，选取左方 U11 及 U10 将 DGND 的管脚选入，后面在 ADS 中作为参考地使用。最后移除 L9 <1>，这个管脚在本次仿真中不会用到。完成后如图 13-4 所示。

(5) 自动添加端口并导出所选区域版图：切换窗口至“5) Ports”栏。勾选“Do not add negative ref. pins”及“Combine on discrete(R,L,C...)”。接着点选“AutoPlace”生成端口。完成后如图 13-5 所示。

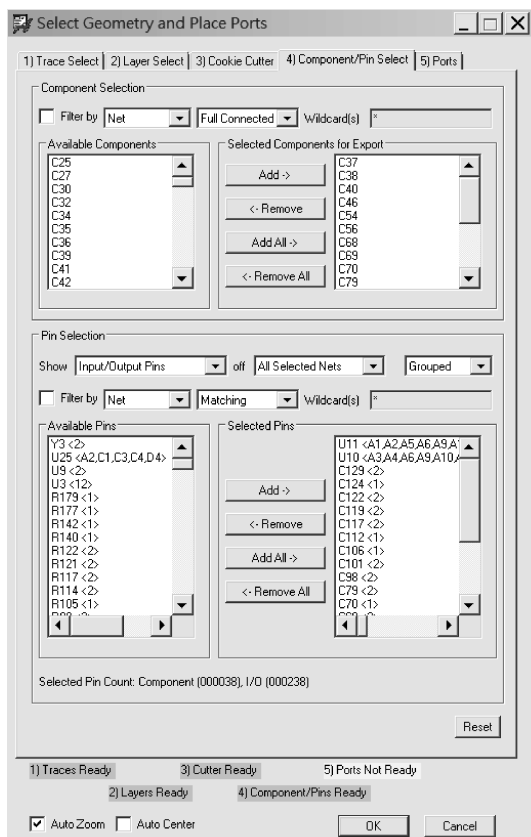


图 13-4 Allegro 版图导出——元件及管脚选择

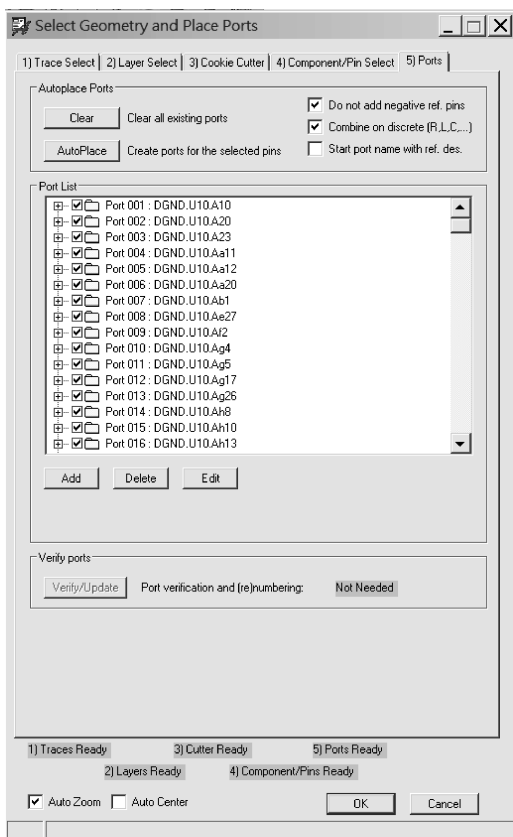


图 13-5 Allegro 版图导出——添加端口

单击“OK”按钮完成所有的设定。最后选取“Export to ADS/EMPro”中的“Export Selected→To ADS”导出所选区域的版图，如图 13-6 所示。

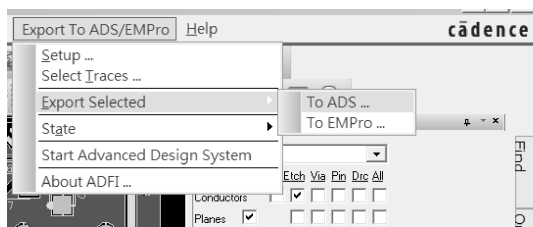


图 13-6 导出选定的 Allegro 版图区域



## 13.2 将 ADFI 导出的版图导入到 ADS

(1) 启动 ADS，并建立一个新的 workspace，如图 13-7 所示。

(2) 在 ADS 软件主窗口中，鼠标单击“File→Import→Design”开启 Import 窗口，如图 13-8 所示。文件类型 (FileType) 项下选择 ADFI File Format，并使用文件浏览框找到导出的 adfi 文件，单击 OK 按钮，并在 Option 选项中使用默认设置，如图 13-9 所示，再次单击 OK 按钮开始导入。导入到 ADS 中的版图如图 13-10 所示。

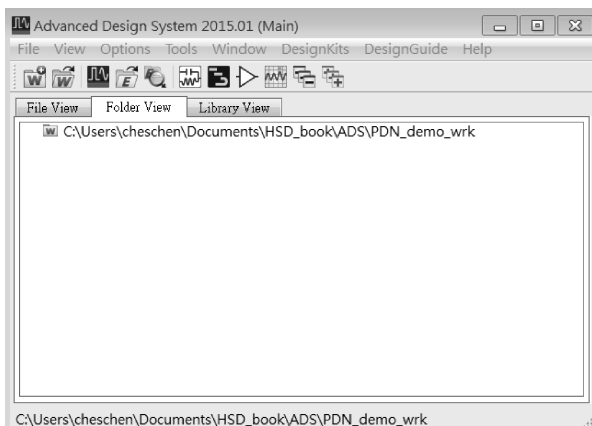


图 13-7 建立新的 ADS 工程

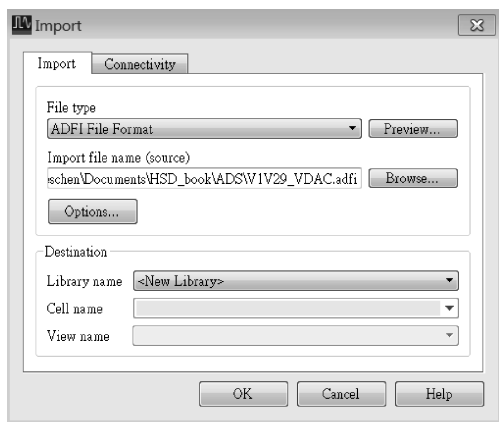


图 13-8 导入 ADFI 设计

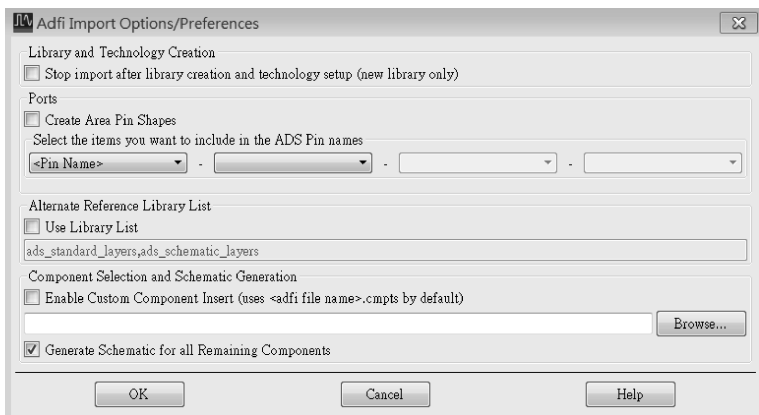


图 13-9 导入 ADFI 选项设置

(3) 整理导入的版图：导入的版图基本上是由多边形 (Polygon)、圆形 (Circle) 及走线 (Trace) 组成的。在版图的某些区域，这些形状叠摞在一起并没有形成一个整体，如图 13-11 所示 Top 层的一块版图。

可以对版图进行处理，合并所有连接在一起的形状，以加快软件的处理速度。

首先将 Trace 转换为 path：在版图窗口中的 Navigator 工具窗口中（可以通过版图窗口 View→Navigator Window 激活此工具），使用鼠标选取所有的 Trace，如图 13-12 所示。

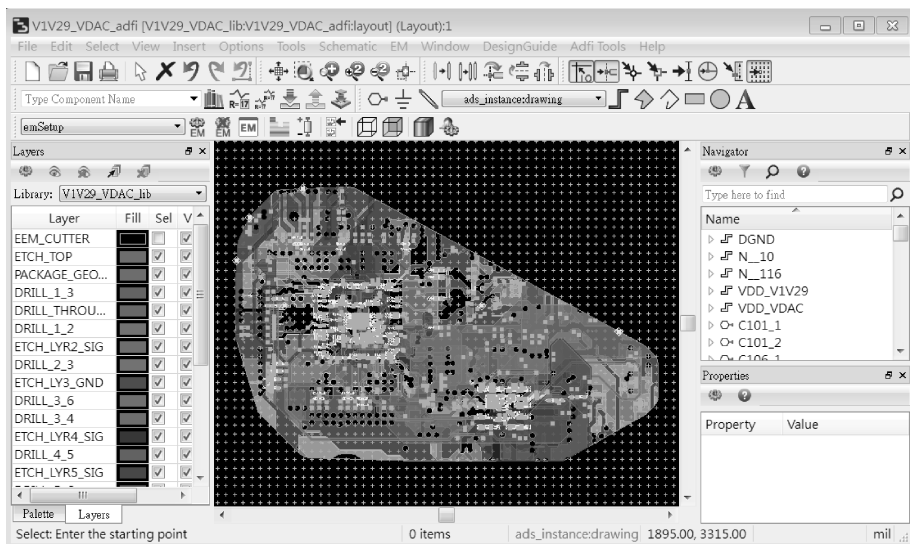


图 13-10 导入到 ADS 中的版图

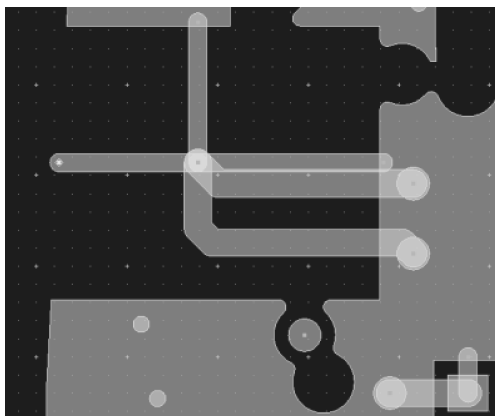


图 13-11 导入的版图局部区域

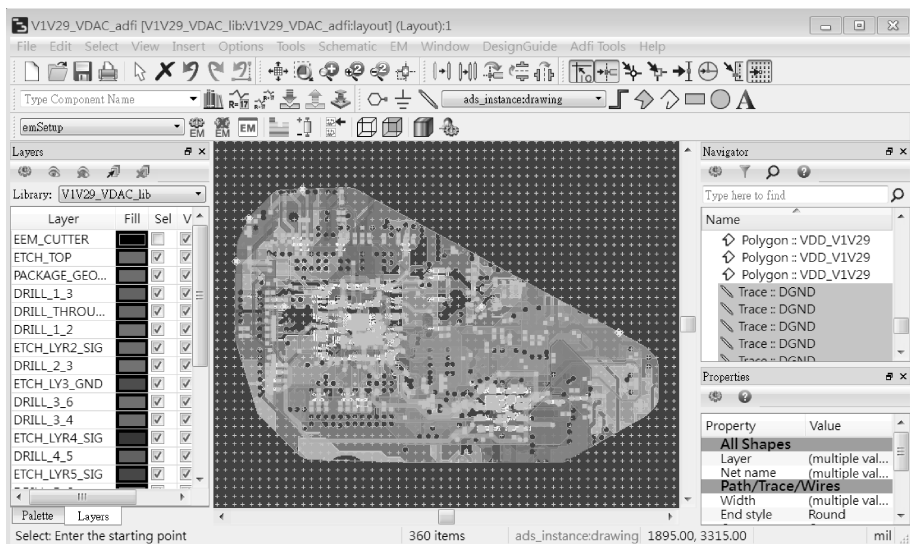


图 13-12 在 Navigator 窗口中选择所有的 Trace

之后，单击版图菜单“Edit→Path/Trace→Convert Trace to Path”，如图 13-13 所示。

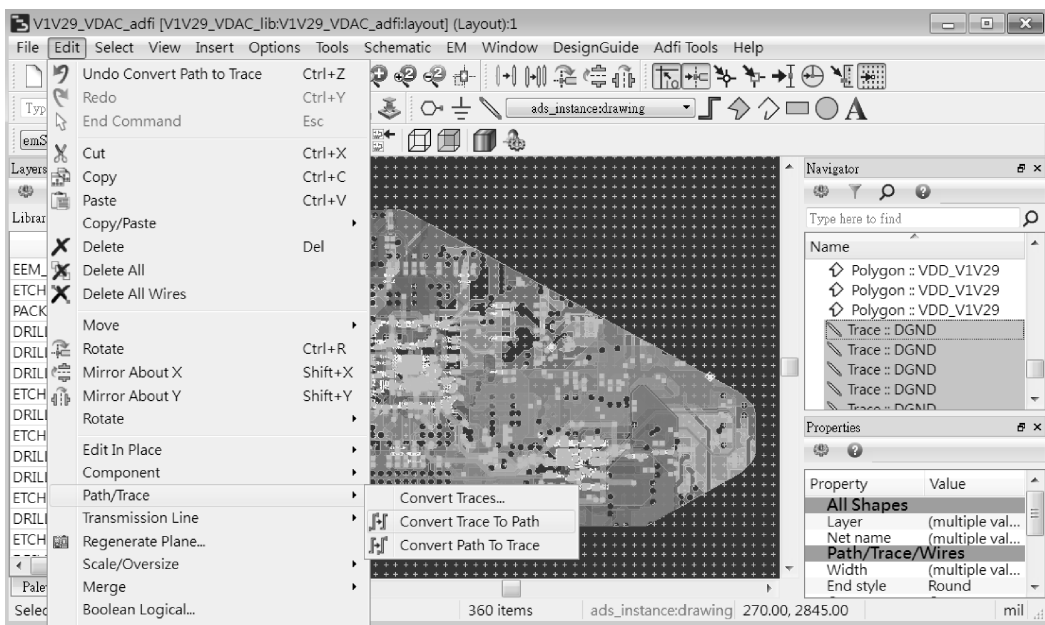


图 13-13 使用编辑功能将 Trace 转换为 Path

转换完成后，会有一些 Trace 无法转换，暂时忽略。

接着选取从 Circle 至 Polygon 所有的形状，并使用菜单“Edit→Merge→Union”功能将相连的形状合并在一起。完成之后如图 13-14 所示。

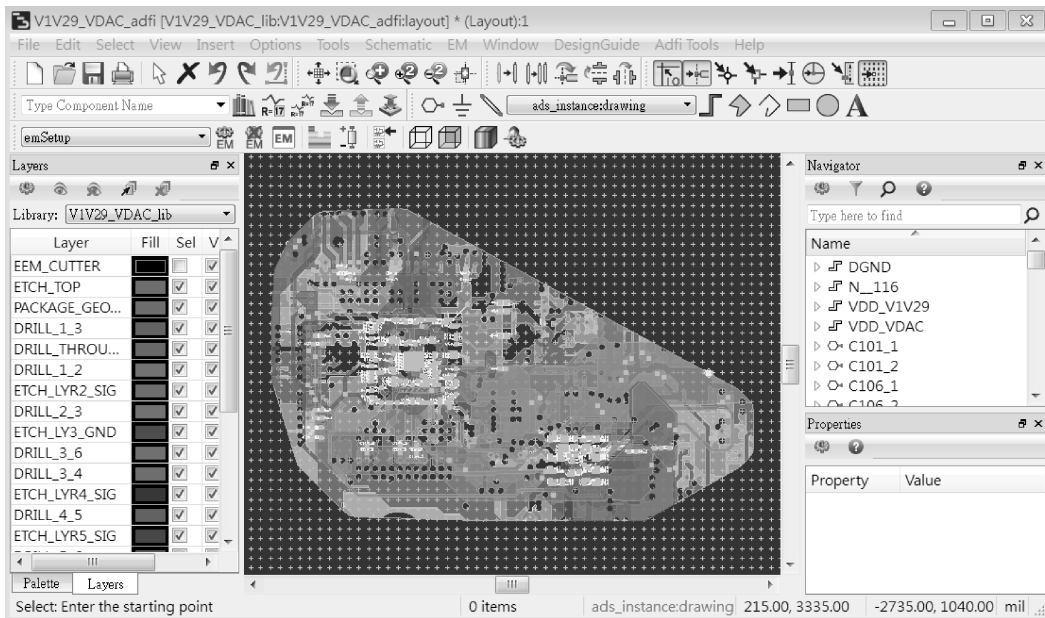


图 13-14 整理之后的版图



之前看到的 Top 层的区域版图也简明多了,如图 13-15 所示。

(4) 删除无用的碎片:导入的版图中还有一些无法转换的 Trace,其中 N\_116 就是一些。在 Navigator 窗口,将 N\_116 展开,选取 Trace::N\_116 并将其删除,如图 13-16 所示。

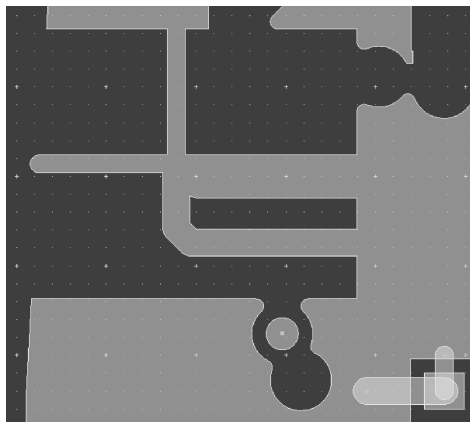


图 13-15 整理之后的版图线条明显减少

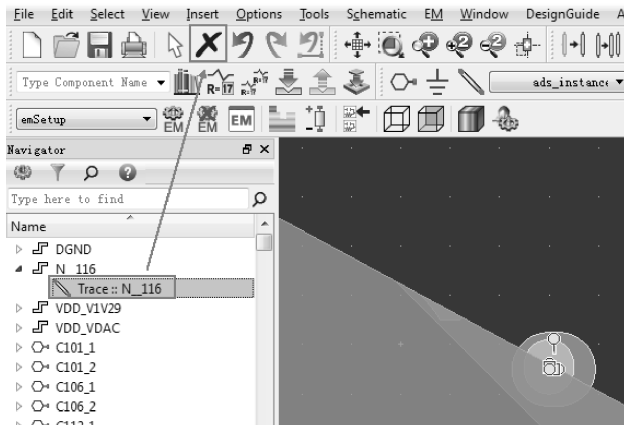


图 13-16 删除无用的走线及网表

在 ADS2015 中,如果版图形状连接至特定的网表,在其形状之后会注明其连接的网表名称,若版图形状与网表连接,则注明其坐标,如图 13-17 所示。

这些没有与网表连接的形状通常都是一些碎片,如图 13-18 所示的 Top 层上的一些形状。

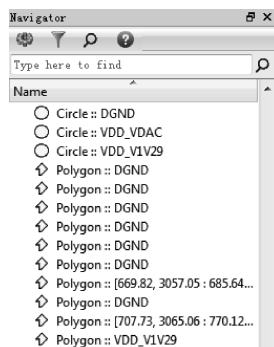


图 13-17 版图形状及其连接的对象

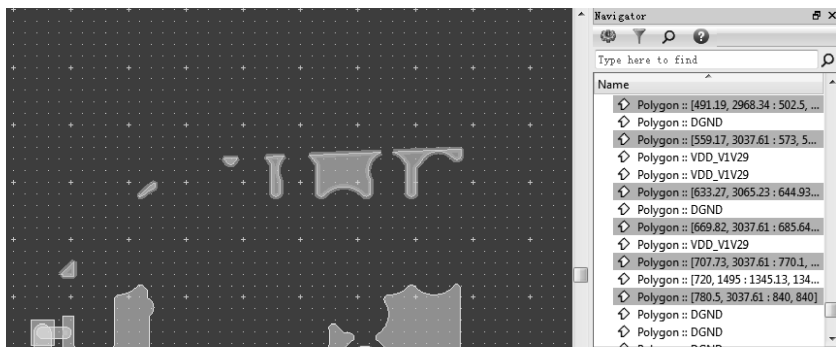


图 13-18 版图放大后,没有连接到网表上的形状

将这些碎片形状删除,不会改变任何版图的电气连接关系,但会加速 ADS 软件的电磁场仿真速度。在 Navigator 的搜寻窗口中输入“::[”将没有转接至网表的形状筛选出来,如图 13-19 所示。

选取所有的形状并将其删除。最后将 Navigator 中的“::[”清除,完成后如图 13-20 所示。

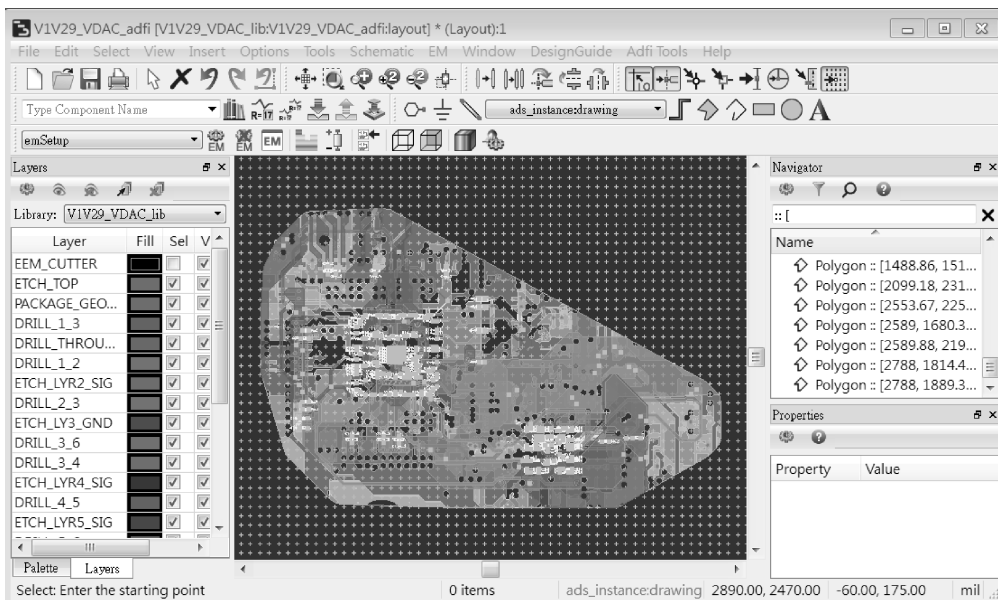


图 13-19 在 Navigator 窗口中选择所有没有连接到网表上的形状

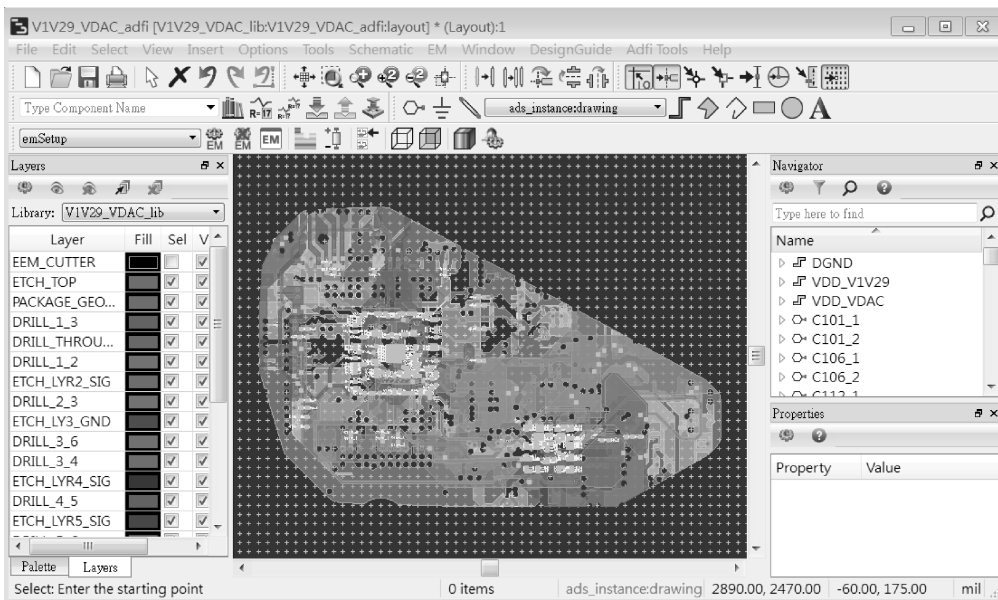


图 13-20 经过整理后的版图



### 13.3 EM setup 设置

(1) 检查层叠 (substrate) 设置: 打开层叠设置窗口, 检查层叠设置与材料参数是否与事实相符, 如图 13-21 所示。

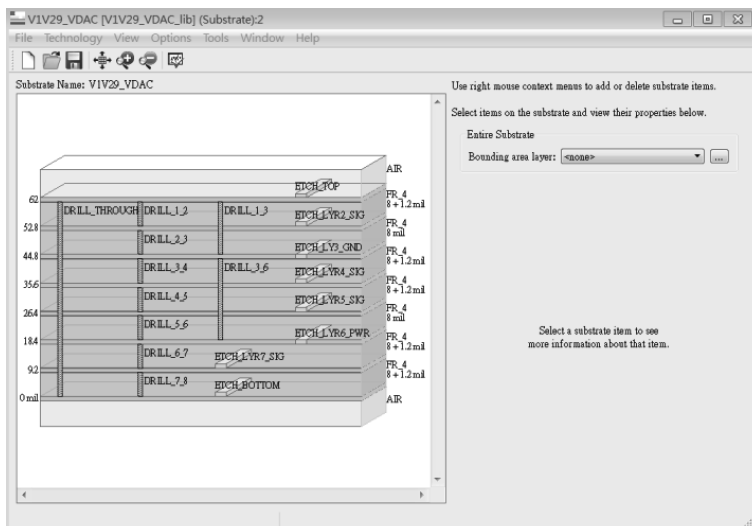


图 13-21 层叠设置窗口

一般而言，下层金属的扩展方向与上层相反，可将层叠设置修改成图 13-22 所示。

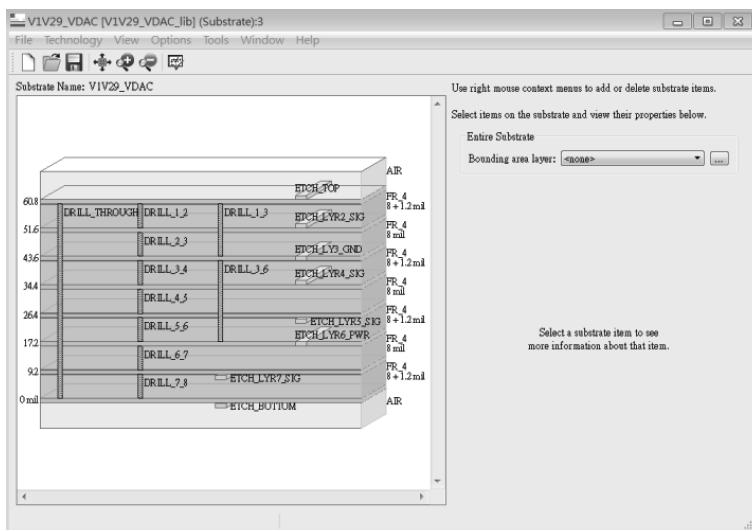


图 13-22 对下层金属的扩展方向进行修改

(2) 修改 EM 设置：在 EM 设置窗口中，将“Setup Type”修改成“EM Simulation/Model”模式，如图 13-23 所示。

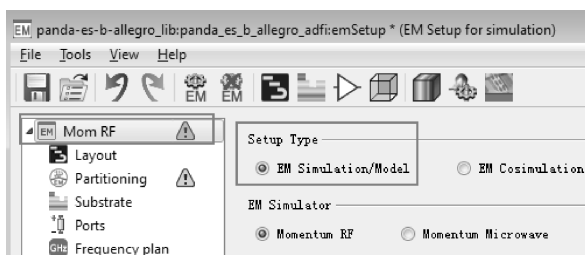


图 13-23 设置电磁场仿真类型

鼠标单击“Ports”项，之后单击“Edit”按钮打开 Edit 窗口。在上方的 S-parameter Ports 窗口中选择所有 U10\_及 U11\_的端口，并单击删除按钮进行删除，如图 13-24 所示。

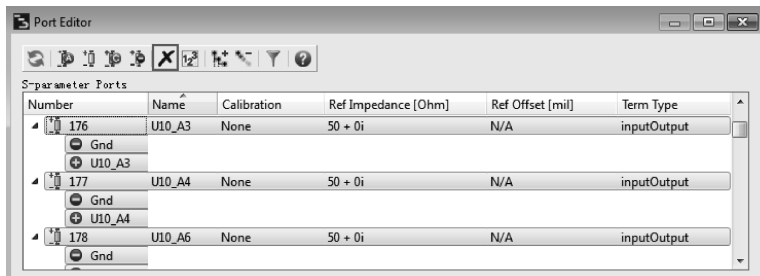


图 13-24 删除 U10\_及 U11\_默认端口映射

完成后如图 13-25 所示。

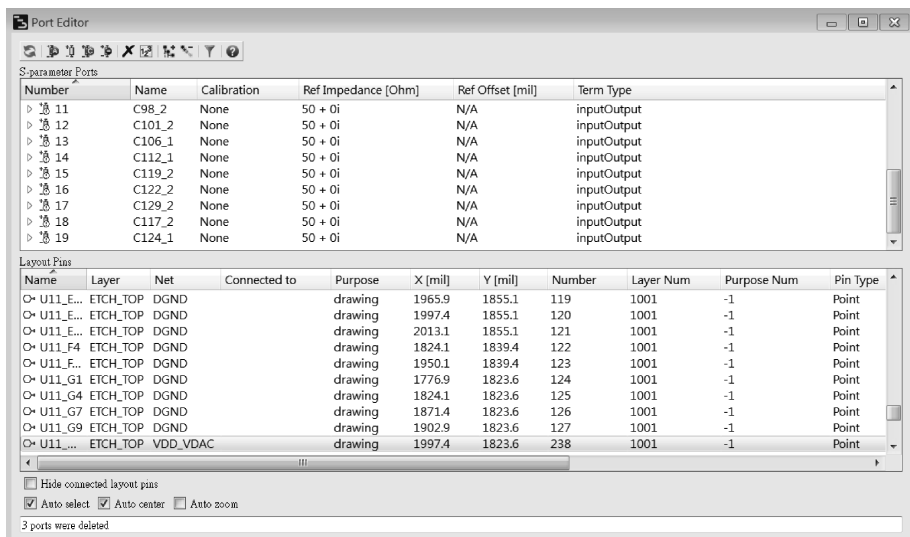


图 13-25 删除 U10\_及 U11\_端口映射后，只留下电容的端口映射

点选下方的“Hide connected layout pins”功能，并点选 Layout pins 中的“Net”排序。选取 U10\_A11 及 U10\_G12 后单击右键，选取“Create a port with clustering pins”建立一个 port，如图 13-26 所示。

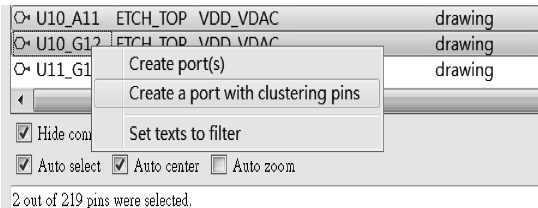



图 13-26 为 U10\_中连接 VDD\_VDAC 网络的管脚创建组合端口

之后选取 U11\_G15，选取“Create port(s)”，建立一个 port。对于所有连接到“VDD\_V1V29”网表上的端口也使用相同的方式建立两个 port：所有以 U10\_开头的，选取“Create

a port with clustering pins” 建立一个 port，以 U11\_开头的，选取 “Create port(s)”，建立一个 port。完成后如图 13-27 所示。

使用  快捷按钮展开 Port18(U10\_A3)，可以发现所有的管脚皆接在正端，这表示是计算 S 参数时，所有的管脚都是同电位的。

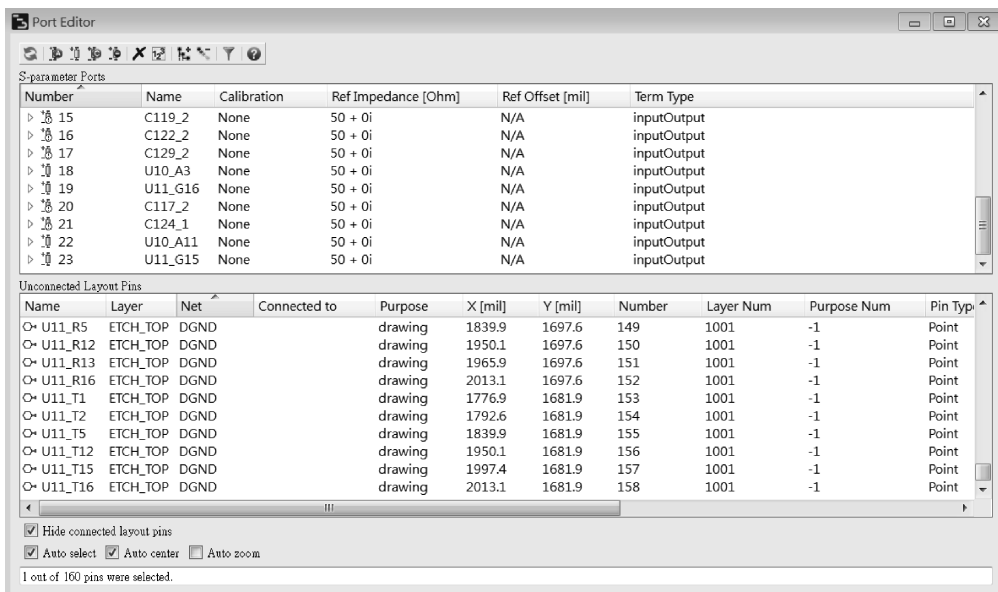


图 13-27 重新对 U10\_和 U11\_管脚进行端口映射

接着在窗口下方选取以 U10\_开头的所有连着 DGND 的管脚，并将其拖曳至 Port18 的负端，完成后如图 13-28 所示。

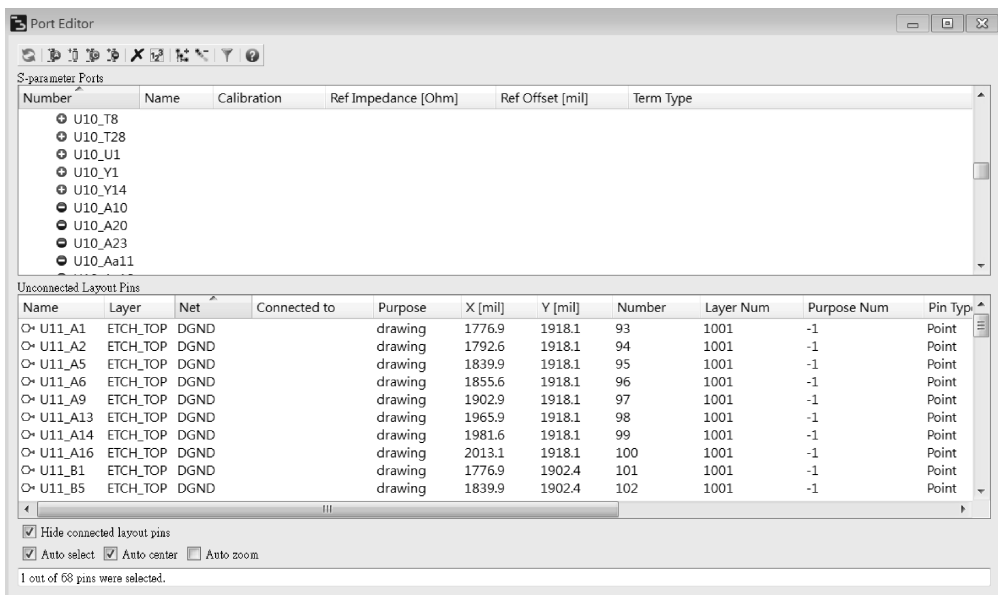


图 13-28 将 U10\_ DGND 管脚映射为地参考端口

将窗口下方其余的管脚加入 Port19(U11\_G16) 的负端。接着取消“Hide connected layout pins”功能，窗口中会显示每个管脚连接至哪个 port 的正端或是负端，如图 13-29 所示。

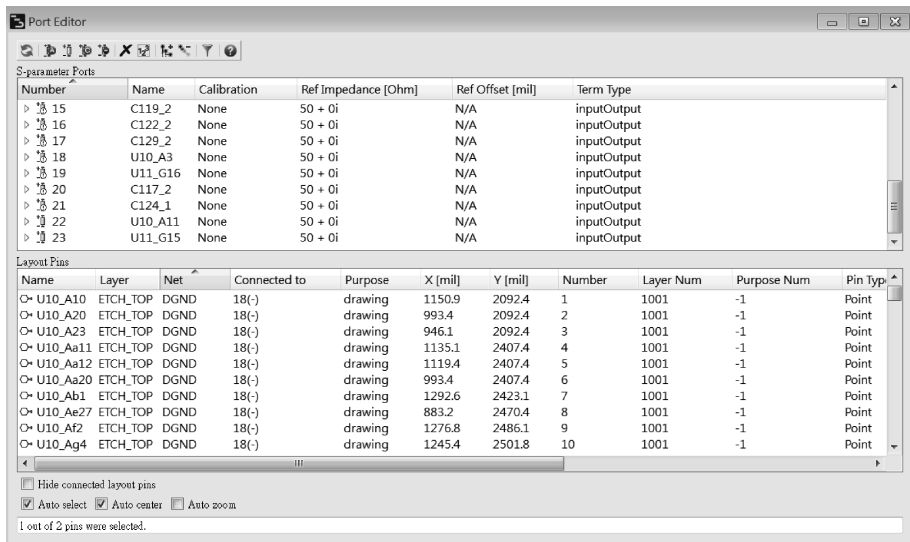


图 13-29 下方的版图管脚窗口中显示该管脚映射的端口信息

再次将窗口下方以 U10\_开头的连接到 DGND 的管脚加到 Port22(U10\_A11) 的负端，以 U11\_开头的连接到 DGND 的管脚加到 Port23(U11\_G15) 的负端。最后将 P18、P19、P22、P23 的参考阻抗设为 10ohm 便完成 Port 设定。完成后如图 13-30 所示。

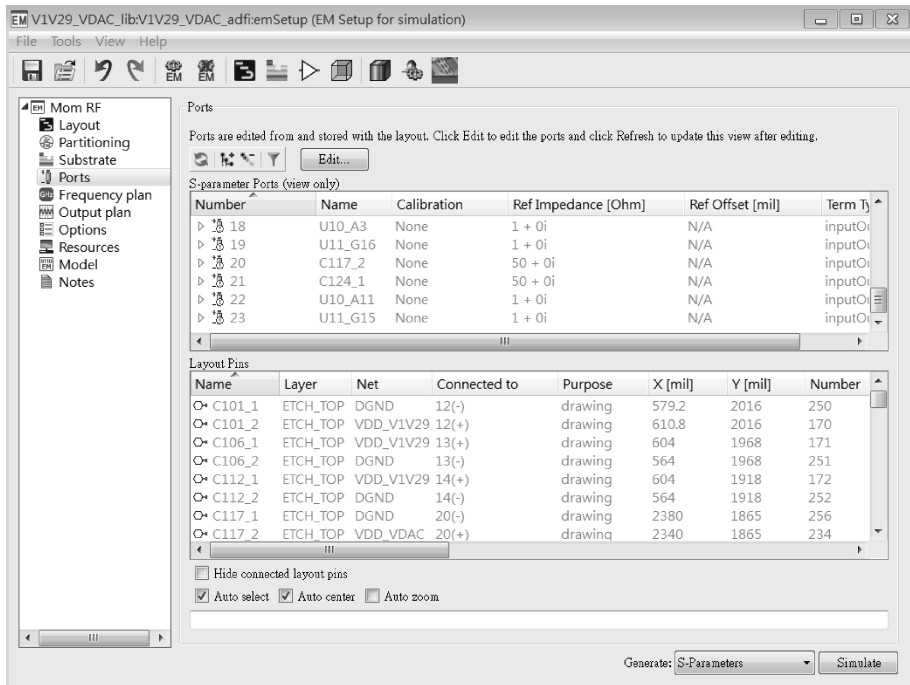


图 13-30 为电源网络端口设定参考阻抗

(3) 设定电磁场仿真频率范围及仿真选项：单击窗口中的“Frequency Plan”选项，将“FStop”改为 0.3GHz。一般而言，在电路板上的电源完整性关心的频率约为 300MHz 以下。

如图 13-31 所示，之后单击窗口中的“Options”选项，在 Physical Model 栏下的 Global 选项中将 Thick conductor 修改为 Sheet，将 Via 修改为 Lumped。这是仿真速度最快的设定方式，若要获得更精确的结果，可以使用“Thick conductor = 2D Distributed”及“Via = 2D Distributed”的设定。但相对仿真时间及所需的内存会比较多。其余的设定保持默认值即可。

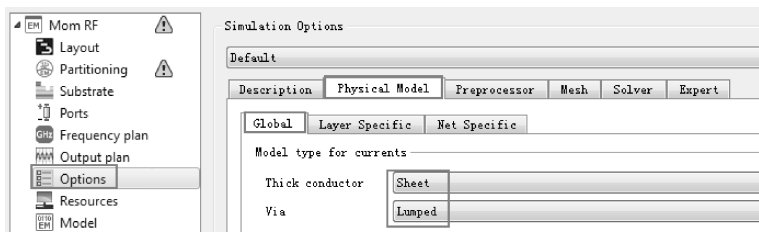


图 13-31 设置仿真选项

(4) 运行 EM 仿真：在进行 EM 仿真前，可将窗口最右下方的 Generate S-parameter 修改为 Generate Mesh 估计 EM 仿真所需的内存，如图 13-32 所示。如果所需的内存太大，可修改 Options 中的设定来减少内存的使用量。

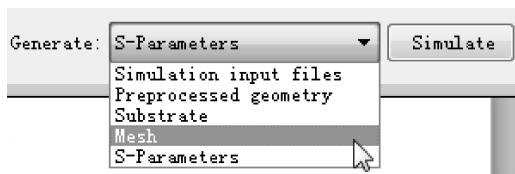


图 13-32 运行版图网格剖分有助于了解仿真需要的计算资源

在运行仿真之前，注意到仿真设置页面中包含黄色的警告标志，说明仿真设置上有需要修正的地方，如图 13-33 所示。

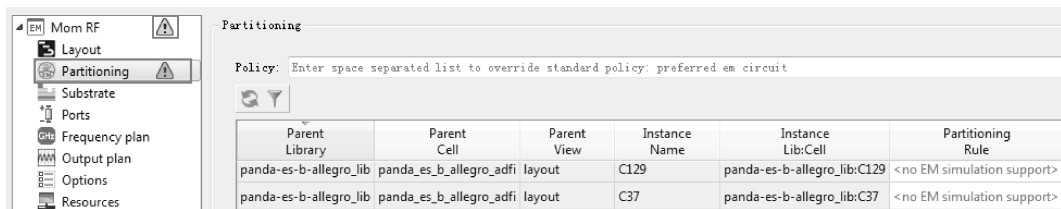


图 13-33 仿真设置中包含警告信息

从 Allegro 导入的版图中包含了众多的去耦电容元件，到目前为止，我们并没有为这些去耦电容指定仿真模型。在本实例中，只需要对导入的版图进行电磁场仿真得到基本布线的模型就可以，所以只需要在版图窗口中删除这些端口就可以了。

在版图窗口中的 Navigator 栏中，选中所有 Cxxx 开始的单元设计 (Cell)，并进行删除，如图 13-34 所示。

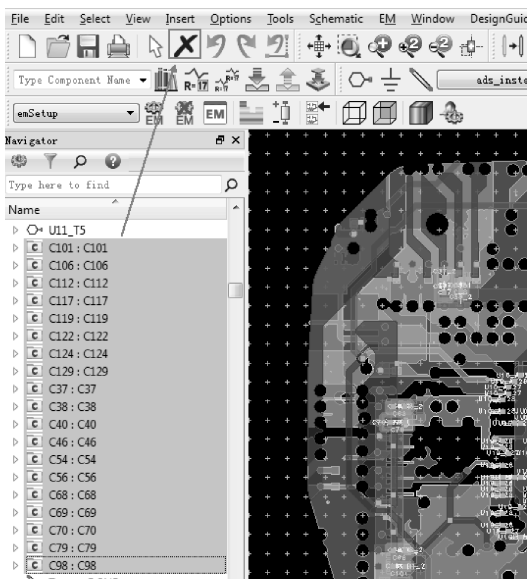



图 13-34 选择所有 Cxxx 开头的单元，并进行删除

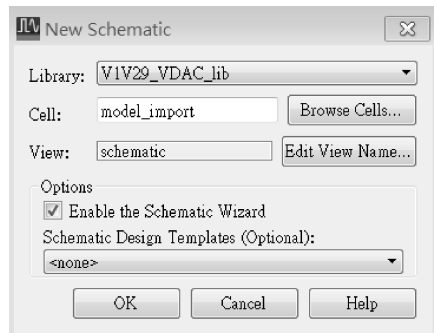
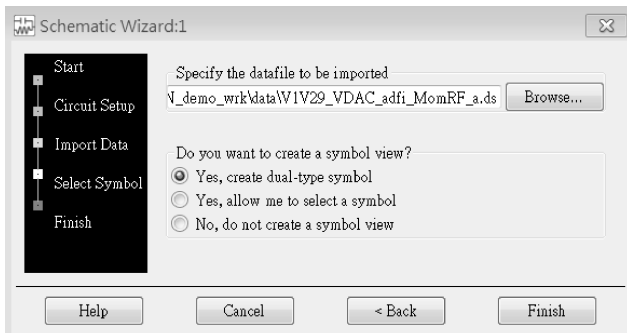
回到电磁场设置页面，可以看到原来的警告信息已经消失。将页面右下角修改为 Generate S - parameter 后单击“Simulate”运行电磁场仿真（或者单击快捷图标）。



## 13.4 PDN 阻抗响应


(1) 导入仿真结果：创建一个新的 schematic view 命名为“model\_import”，并打开“Enable the Schematic Wizard”功能，如图 13-35 所示。

在仿真向导页面中选择工作区下 data 目录下的\_a.ds 档案。EM 仿真完的结果会产生两个数据文件，一个是原始仿真使用的频率点下的 S 参数结果，另一个是 Adaptive Frequency Sweep（AFS - 自适应频率扫描）产生的频率点下的 S 参数结果。我们选用 AFS 仿真的结果，如图 13-36 所示。

图 13-35 创建新的原理图并  
激活原理图向导工具图 13-36 指定仿真数据并  
建立原理图符号



完成之后 ADS 自动产生一个 SNP 组件及指定所有的 pin。将 Ref 的 pin 删除并加入 Ground symbol，如图 13-37 所示。

接着打开 symbol view，单击左方的  按钮重新产生 symbol，完成后如图 13-38 所示。

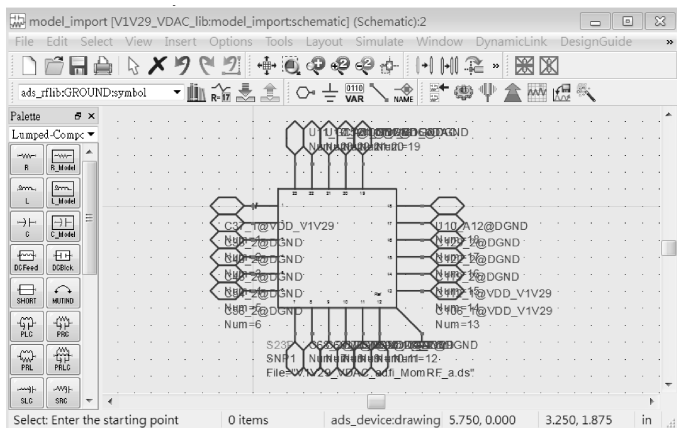


图 13-37 使用向导工具自动生成的原理图

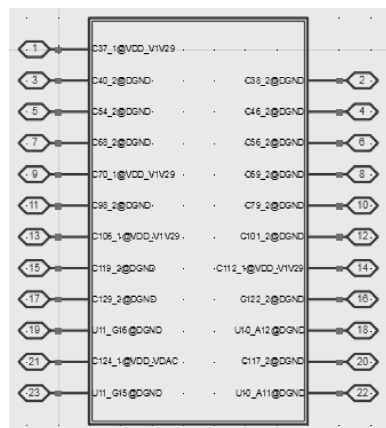


图 13-38 自动产生的原理图符号

(2) 建立 S 参数仿真模板：创建新的原理图“TB\_PDN”，并将上面步骤中建立的“model\_import”子电路加入到这个原理图中。其中 U10 代表从芯片看出去的电源网络，U11 代表从电源模块看出去的电源网络。将“Term”加在 U10 的两个 pin 上，U11 的两个 pin 接 Ground。接着将“Term”的参考阻抗设定为 10 Ohm，如图 13-39 所示。

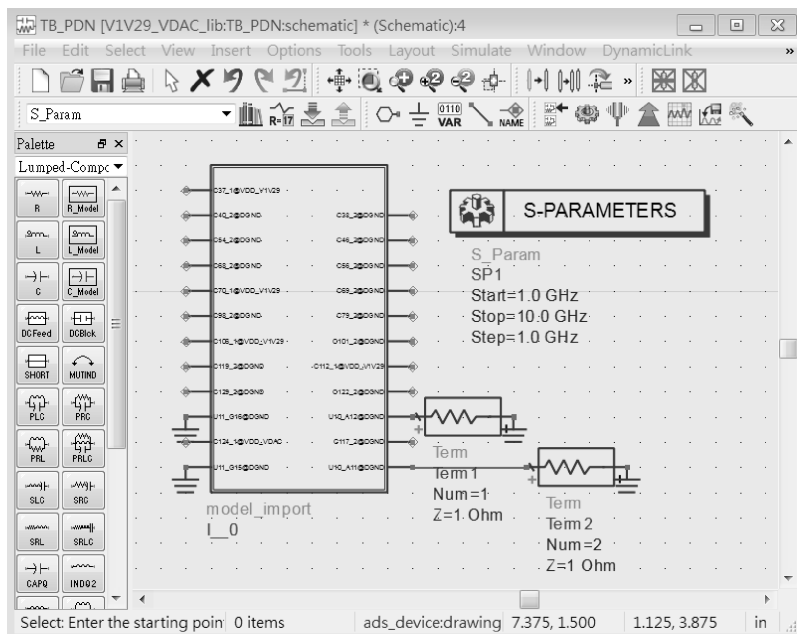


图 13-39 S 参数仿真原理图

将 S 参数仿真控制器中的频率范围设定为 1Hz ~ 0.3GHz、log sweep、10pts/decade。在 Parameter 中打开 Z-parameters 的功能，如图 13-40 所示。

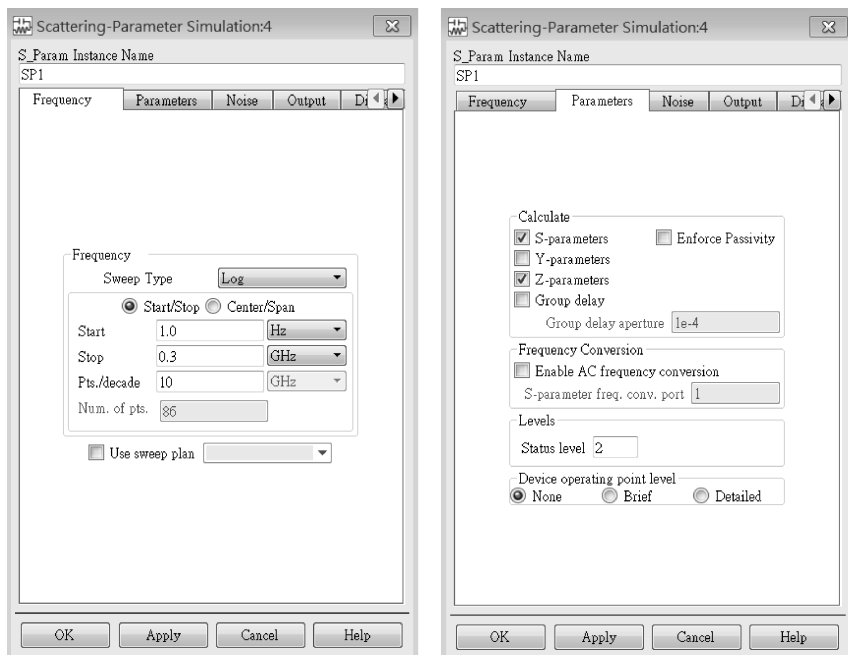


图 13-40 S 参数仿真设置

(3) 绘制 PDN 的阻抗曲线：运行仿真，在数据显示窗口中画出  $Z(1,1)$  及  $Z(2,2)$  的幅度曲线，并使用对数方式显示 X 轴及 Y 轴，如图 13-41 所示。

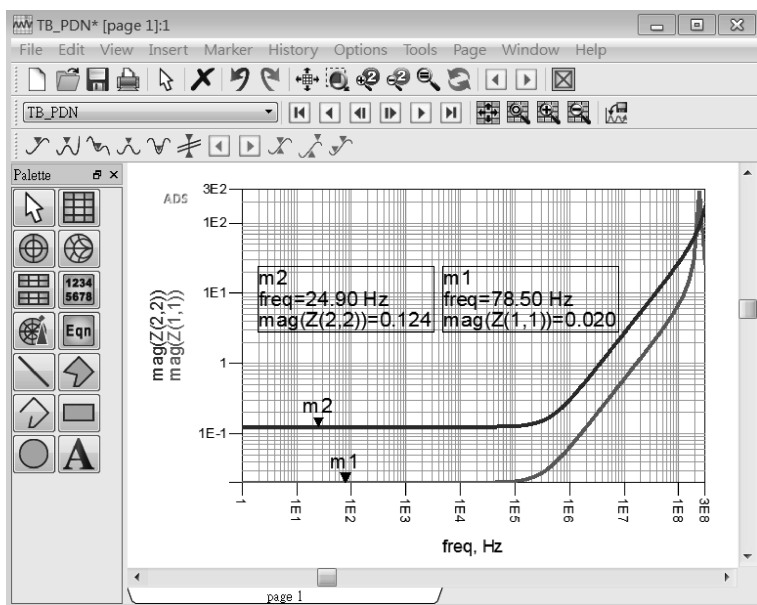


图 13-41 电源分配网络阻抗曲线

使用标记可以读出所关心频点处的阻抗。

### 【总结】

本实例示范了如何从 PCB 版图设计中提取 PDN 网络导入到 ADS 软件中进行 S 参数仿真，进而计算其电源分配网络的阻抗曲线。

# 附录 A ADS 信号完整性仿真 常见问题及解答

【问题 1】我是信号完整性仿真的新手，该如何入手学习使用 ADS 进行信号完整性仿真？

图 A-1 所示的信号完整性仿真矩阵图可以帮助快速定位您的信号完整性问题。

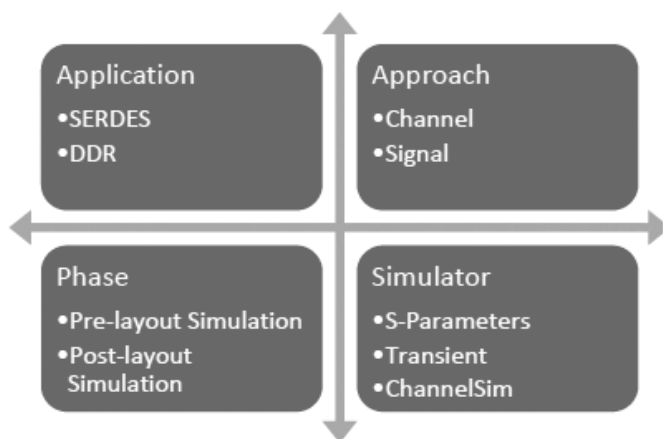


图 A-1 信号完整性仿真矩阵图

首先，要明确您的设计应用是串行总线（SERDES）还是并行总线（如 DDR）技术。这两类设计的信号完整性仿真技术有很大的区别。在 SERDES 设计中，通道的损耗对系统性能影响最大，通常需要使用均衡器补偿通道的损耗。在 DDR 设计中，延时、负载和串扰是最主要考虑的问题，时序分析在 DDR 分析中是必须要做的。

其次，检查信号完整性问题有两种途径：

- ☺ 按照通道特性，研究通道的损耗、阻抗失配及串扰。
- ☺ 按照信号特性，分析信号的波形、眼图等，评估信号质量。

再次，信号完整性仿真通常分为两个阶段，即版图前仿真（简称“前仿”）和版图后仿真（简称“后仿”）。“前仿”阶段主要制定 PCB 版图规则，这一阶段主要使用简单的传输线模型搭建仿真通道；“后仿”阶段检查电路布线后的性能，这一阶段需要通过电磁仿真方法提取实际电路布线的通道模型。

最后，要了解三种不同电路仿真器的用途。S 参数仿真器用于计算通道的频域特性；瞬态仿真器用于仿真通道的时域波形；通道仿真器用于计算通道的统计眼图特性。

图 A-2 所示的是采用通道仿真器仿真的 SERDES 链路模型，从中可以查看眼图沿通道的变化。图 A-3 所示的是使用电磁场求解器得到的 SERDES 通道的 S 参数示例。

【问题 2】串行总线（SERDES）与 DDR 总线在信号完整性仿真上有什么区别？

SI 仿真与分析是 SERDES 和 DDR 通道设计的重要组成部分。由于在拓扑、IO 结构和时间上的差异，SI 设计 SERDES 和 DDR 之间可以是非常不同的。例如，串行总线信号完整性

设计的主要瓶颈是信道带宽。因此，阻抗和插入损耗控制是串行总线通道设计的第一要务。串行总线使用去加重和均衡器改善眼图眼高。至于 DDR，所有 DQ 信号都与 DQS 信号的时序相关，而时钟信号用于对齐命令和地址信号。因此，时序控制是最重要的设计目标。此外串扰和同步开关噪声（SSN）也削弱了 DDR 信号。

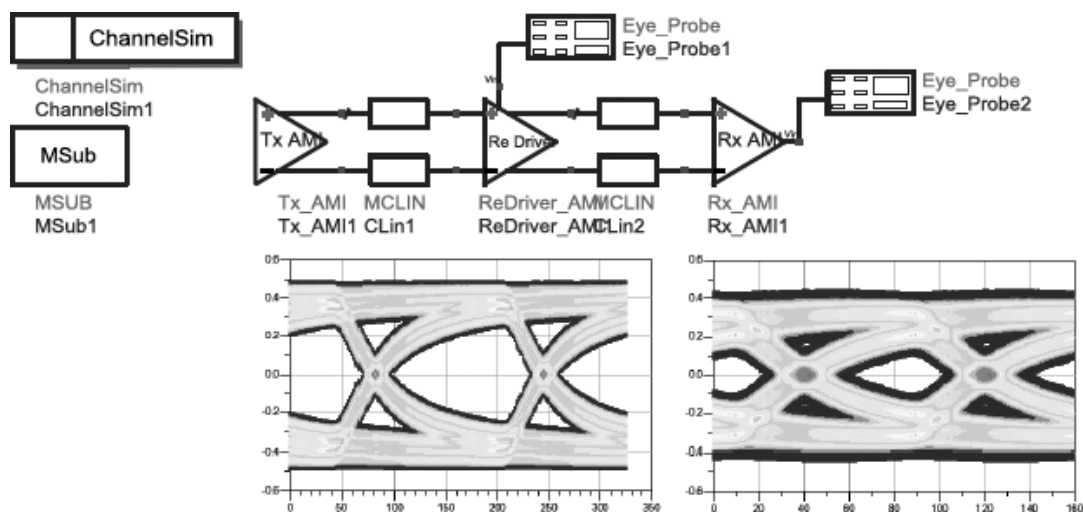


图 A-2 前仿真，眼图随 SERDES 通道位置的变化

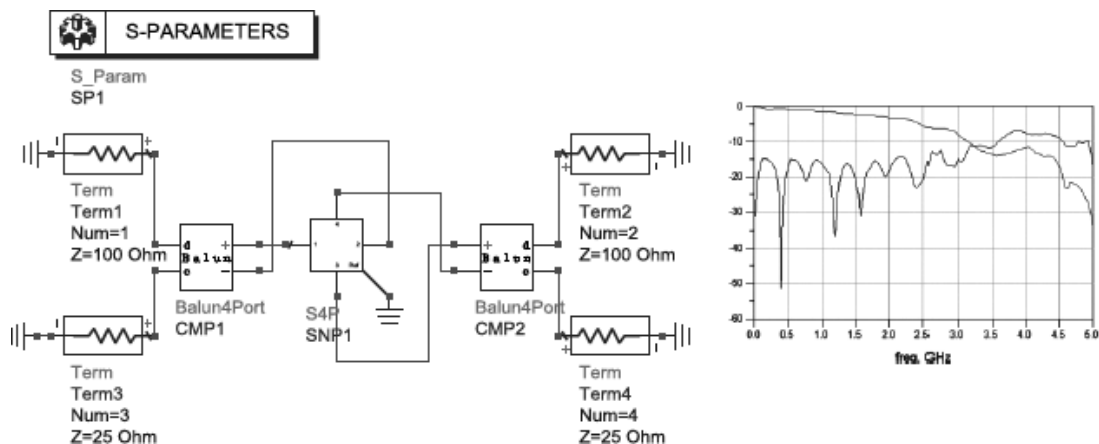


图 A-3 后仿真，SERDES 通道的 SDD 参数

表 A-1 列出了串行总线（SERDES）与 DDR 总线的主要差异。

表 A-1 串行总线（SERDES）与 DDR 总线的主要差异

	串行总线	DDR
应用	PCIE, SATA, USB3...	DDR3, DDR4...
拓扑结构	点到点	点到点, T 分支, Fly - By
信号类型	差分	差分, 单端
端接	固定阻抗	阻抗可调
通道长度	长	较短

续表

	串行总线	DDR
是否编码	是	否
是否均衡	是	否
时钟信号	时钟信号内嵌	外部时钟
开关同步噪声 SSN	弱	强

⑤ 串行总线常用仿真方法：IBIS AMI 模型是一个常见的 IO 模型。通常使用通道仿真器仿真通道的眼图等性能。一般情况，串行总线的通道较长，插入损耗是 SERDES 信号完整性的最关键因素。为补偿信道的高频损失，通常在发射端通过去加重或在接收机使用均衡器，用于打开通道损失闭合的眼图。此外，TDR 仿真和混合模式 S 参数仿真可以快速检查通道质量。

⑥ DDR 常用仿真方法：IBIS 模型广泛应用于 DDR 仿真。通常使用瞬态仿真器计算 DQ 信号在“读”和“写”模式下的建立和保持时间。通过 S 参数仿真在频域中检查插入损耗及电气延迟（群时延）。对芯片的片上端接（ODT）必须由特定控制信号动态调整。通过批处理控制器进行复杂参数扫描，如长度、阻抗、ODT 等。

图 A-4 和图 A-5 分别给出了 SERDES 和 DDR 仿真的典型原理图。

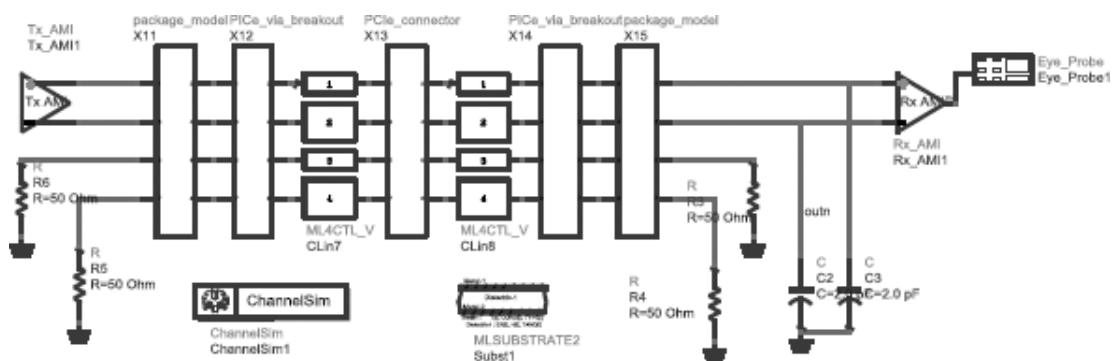


图 A-4 使用 IBIS AMI 模型进行 SERDES 通道仿真

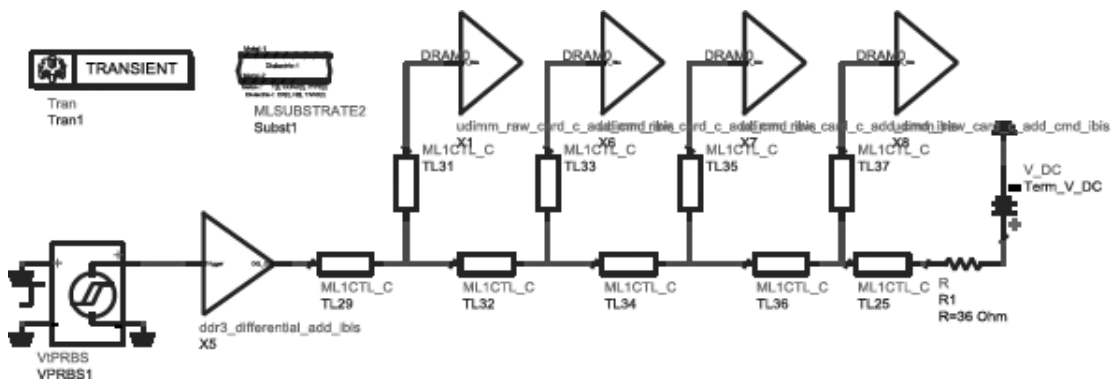


图 A-5 使用 IBIS 模型的 DDR 波形仿真

### 【问题3】ADS 支持哪些信号完整性分析的方法？

有两种类型的 SI 分析方法。

☺ 通道分析：TDR 分析可以帮助识别阻抗不匹配的通道。如果阻抗偏差超过标准，TDR 分析可以帮助进行缺陷定位。S 参数分析提供信道带宽的频域信息。简而言之，通道分析能帮助设计师更深入地了解通道质量。通道分析可获得以下性能：通过 TDR 分析阻抗失配、检查冲击响应、测量带宽、评估延迟、计算插入损耗对串扰比（ICR）等。

☺ 信号分析：SPICE 或 IBIS 模型可作为发射端（Tx）和接收端（Rx）模型。结合 IO 模型和通道模型，运行瞬态仿真检查在接收端的信号波形。信号分析可分析以下性能：检查眼图、评估在特定误码率下的眼高和眼宽、计算建立时间和保持时间、估计峰-峰抖动或均方根抖动、估计误码率与浴盆曲线等。

一般来说，首先进行通道分析，以确保信道质量；然后进行信号分析，以确认完整的系统性能。

通过信号分析可以提供整体的系统性能。但如果系统性能与规格不符，就难以找出根本原因。相比之下，通道分析有助于找出通道的缺陷，但它不能给出全系统性能指标。

图 A-6 为通道分析的示例。图 A-7 为信号分析的示例。

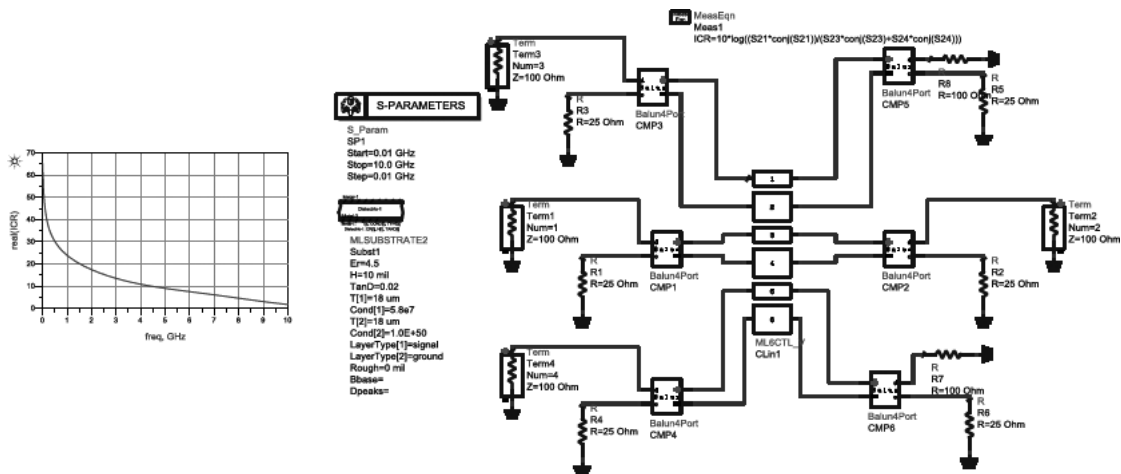


图 A-6 通过通道 S 参数进行 ICR 分析

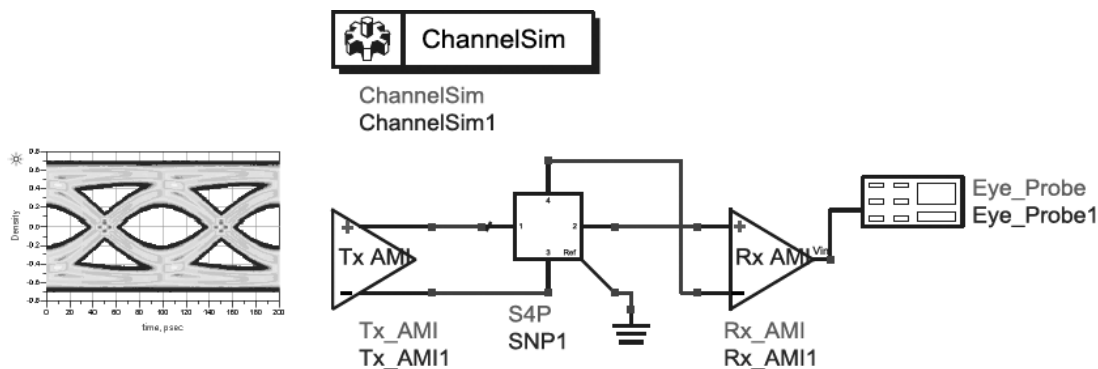


图 A-7 眼图分析

## 【问题 4】版图前仿真和后仿真之间的区别是什么？

PCB 布线对高速数字设计中的信号完整性起着重要的作用。为确保布线后信号的质量，电路板的层叠、阻抗、长度、间距、孔大小、焊盘的大小和电路板材料在 PCB 布线前都必须进行分析。在这一阶段通过前仿真来定义布局布线规则。前仿真阶段的通道模型可以使用简单的基于解析公式的传输线模型。

通常，Layout 工程师将尽可能遵循布线规则进行 PCB 布线。但在现实中，几乎是不可能遵循所有规则的。我们不知道是否不遵循设计规则的部分是否能够正常工作。在版图后仿真阶段，通过电磁场仿真求解器提取实际版图的频域模型。因此，它能够提供更准确的通道模型。

图 A-8 和图 A-9 分别是前仿真和后仿真的示例。

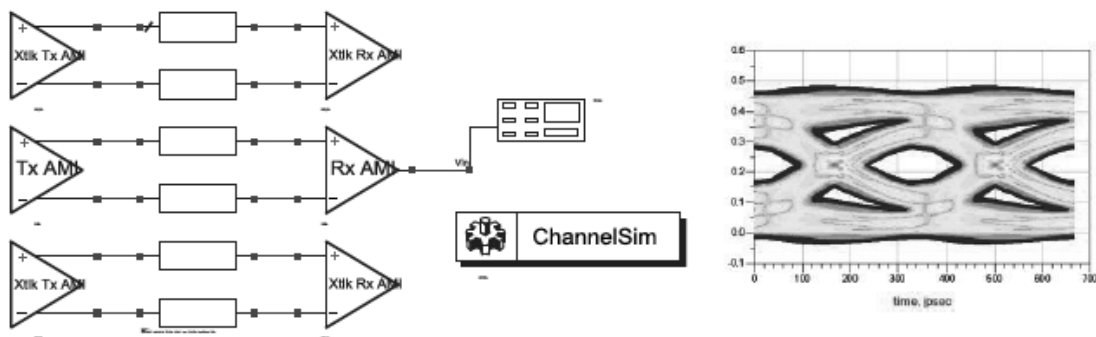


图 A-8 前仿真，使用传输线模型评估眼图

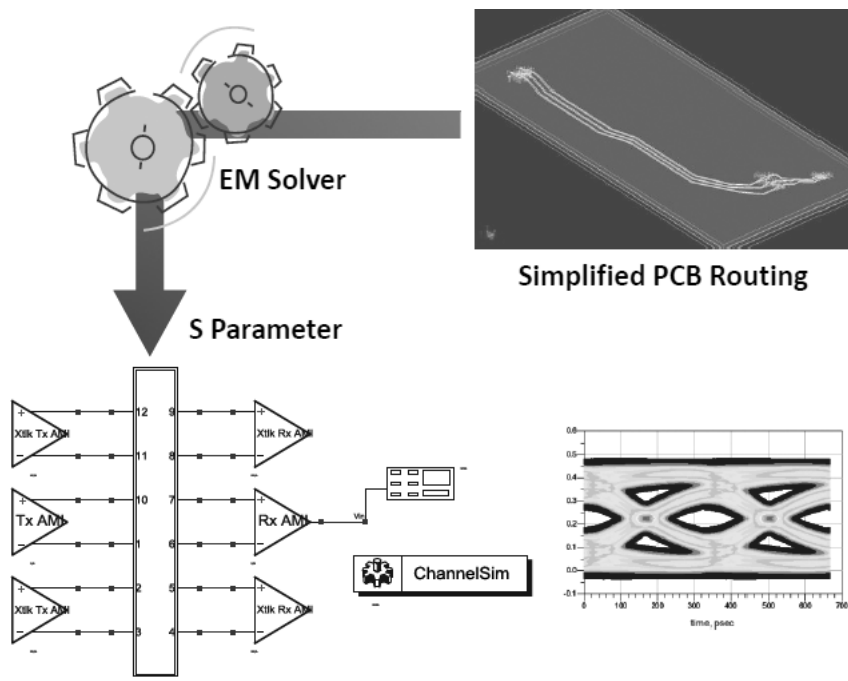


图 A-9 后仿真，采用电磁场仿真的通道模型分析眼图

- ☺ 版图前仿真的特点：使用基于解析公式的简单传输线模型作为通道模型，定义传输线宽、间距等，设计过孔、焊盘等，研究通道拓扑，粗略估计信号性能。
- ☺ 版图后仿真的特点：基于 EM 的传输线模型作为通道模型，计算由于通过过孔、焊盘等引起的阻抗不匹配，检查不理想电流回路的影响，评估其他通道串扰影响，精确估计信号性能。

【问题 5】使用 ADS 进行信号完整性分析需要用到哪些电路仿真器和电磁场求解器？

在 ADS 软件中，有三种电路仿真器用于 SI 仿真与分析。

- ☺ S 参数仿真器：通道 S 参数仿真。S 参数可帮助用户分析插入损耗、带宽和信道的频域串扰。
- ☺ 瞬态仿真器：时间域仿真器，用于分析 TDR、波形上升/下降、振铃和眼图。IO 模型（SPICE 或 IBIS 模型）可以用于瞬态仿真中。
- ☺ 通道仿真器：用于较长的比特序列仿真。通道仿真器比瞬时仿真器效率更高。它用于检查系统的性能，如浴盆曲线、误码率轮廓等。可以在 ChannelSim 中使用 IBIS AMI 或 IO 的行为模型。

ADS 支持以下两种电磁场求解器。

- ☺ 矩量法（MoM）：对于纯平面结构建模，如 PCB 走线、差分对、配电网络（PDN）、螺旋电感等。仿真可以计算频域的 S 参数，可以显示电流分布和辐射图。
- ☺ 有限元法（FEM）：全三维电磁场求解器，用于仿真封装（BGA、引线、键合线）、连接器、插槽等。仿真可以计算频域的 S 参数，可以显示电场、磁场、电流分布和辐射图。

【问题 6】什么是“DesignGuide”，它对信号完整性分析有什么帮助？

对于新手来说，DesignGuide 是上手学习信号完整性分析的一个有用工具。ADS 中有两种 DesignGuide 设计，一种用于射频（RF）电路设计，如放大器、滤波器、混频器等；另一个用于信号完整性分析，如 HDMI、SASII、IBIS AMI 等。我们将介绍用于信号完整性分析的 DesignGuide。

在 ADS 原理图中，从菜单栏列表中选择 DesignGuide。单击 DesignGuide 列表中的项将打开一个 DesignGuide 窗口。例如，单击 USB3 DesignGuide，会弹出如图 A-10 所示的 USB3 设计指南窗口。

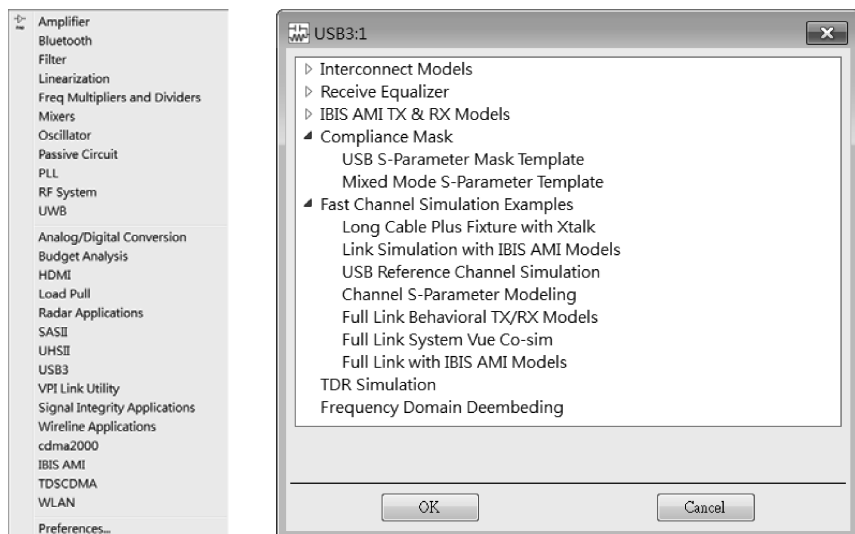


图 A-10 USB3 DesignGuide 界面



DesignGuide 将包括模型、设计模板和数据分析显示。例如，双击 USB3 DesignGuide 窗口中的“USB Reference Channel Simulation”，ADS 会打开“Reference\_Channel\_With\_IBIS\_AMI”的原理图，如图 A-11 所示。这是一个设置 USB3 通道仿真与 IBIS AMI 模型的例子。

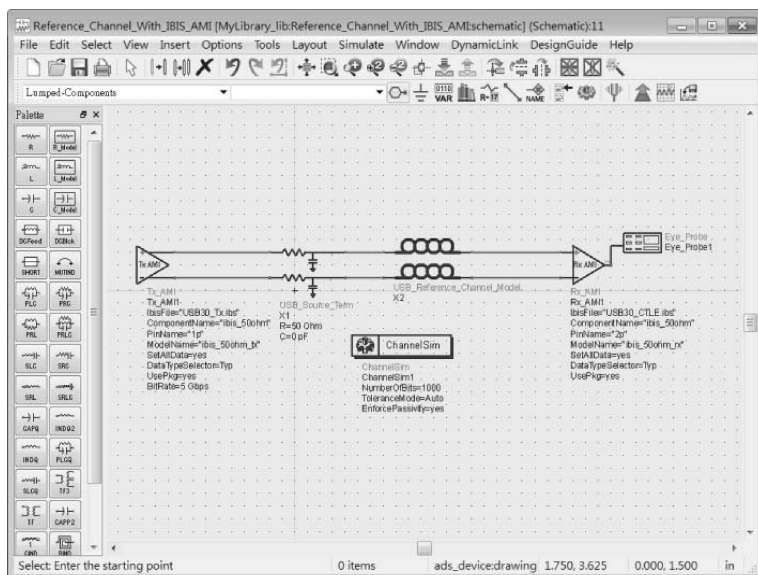


图 A-11 “Reference\_Channel\_With\_IBIS\_AMI” 原理图

若要检查是否信道的 S 参数满足规范要求，请双击“USB S - Parameter Mask Template”加载“USB\_S\_Parameter\_Mask. dds”数据显示文件，如图 A-12 所示。在数据显示窗口中通过等式定义了规范要求的频域模板，以红色曲线显示。我们可以轻松地检查通道是否符合要求。

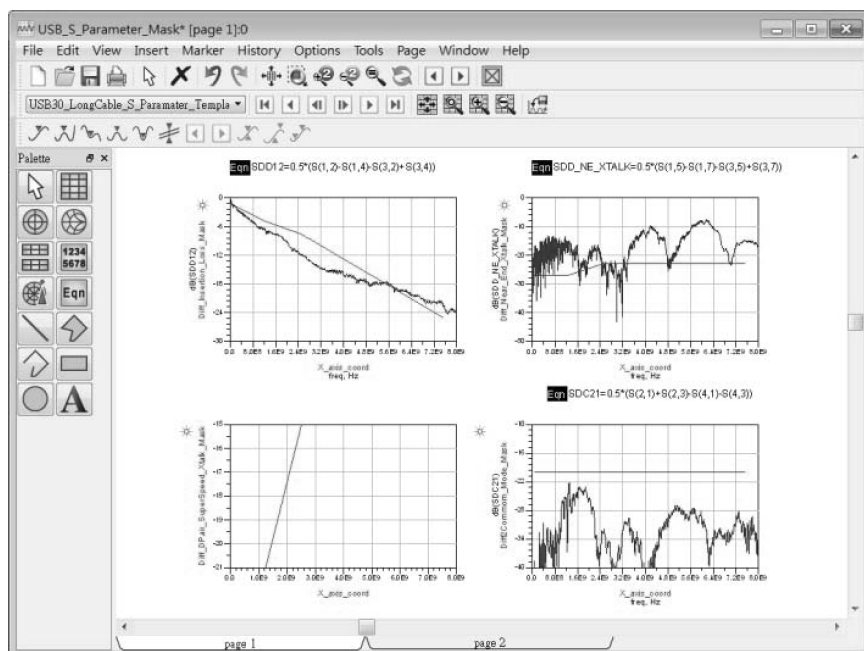
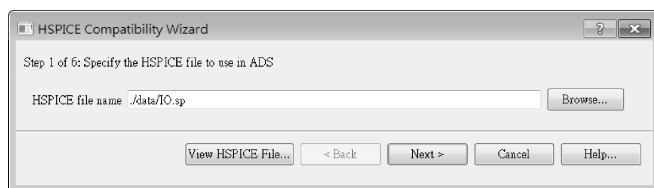


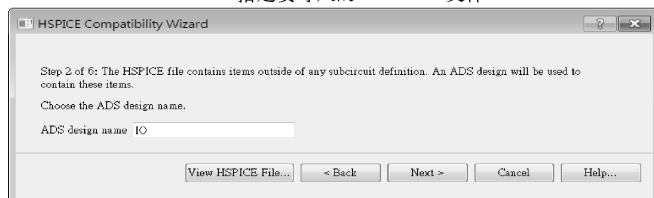
图 A-12 “USB\_S\_Parameter\_Mask. dds” 数据显示文件

### 【问题 7】如何在 ADS 中导入 HSPICE 网络表文件？

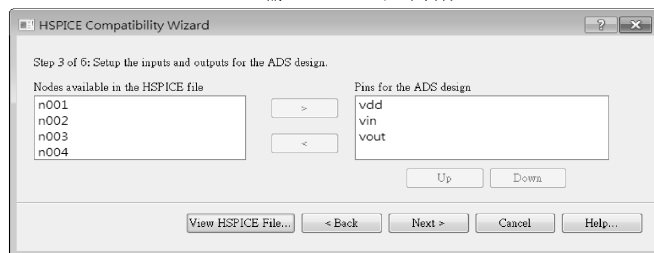
HSPICE 网络表是一种通用格式的网络表和模型文件。要导入 HSPICE 网络表文件，请选择 Tools→HSPICE Compatibility Component→Wizard 打开 HSPICE 兼容性向导窗口，然后按照图 A-13 所示步骤导入网络表。



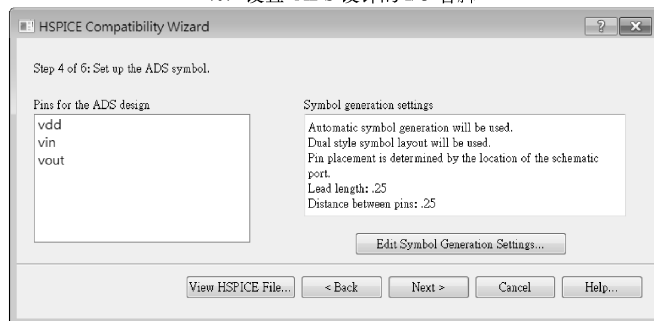
(a) 指定要导入的 HSPICE 文件



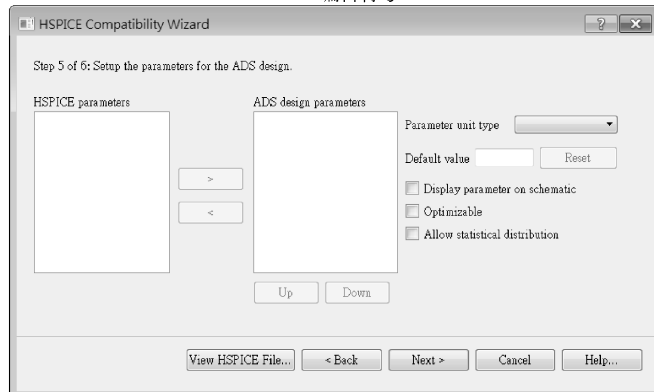
(b) 输入的 ADS 设计名称



(c) 设置 ADS 设计的 I/O 管脚

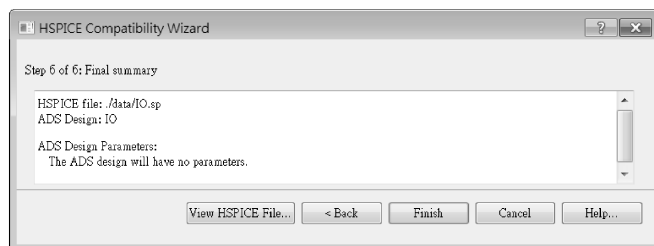


(d) 编辑符号



(e) 设置参数

图 A-13 导入网络表



(f) 查看摘要并单击完成按钮导入网络表

图 A-13 导入网络表 (续)

导入完成后, 可以将 HSPICE 模型与传输线、直流偏置、信号源、眼图探针等元件一起组成仿真原理图, 如图 A-14 所示。必要时, 我们可以进入 HSPICE 模型中, 查看和编辑网络表。

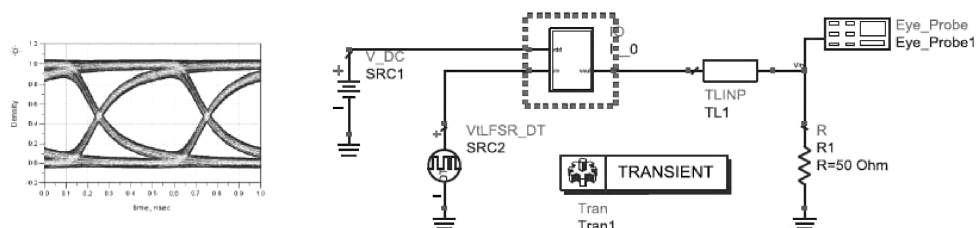


图 A-14 使用 HSPICE 网络表模型进行信号完整性仿真

【问题 8】如何使用 SystemVue 构建串行总线发射端和接收端 IBIS AMI 模型?

SystemVue 是进行电子系统开发 (ESL) 的一款设计工具, 用于系统体系结构工程师设计电子系统的数学模型。SystemVue 可用于 SERDES 体系结构建模与验证。图 A-15 显示了

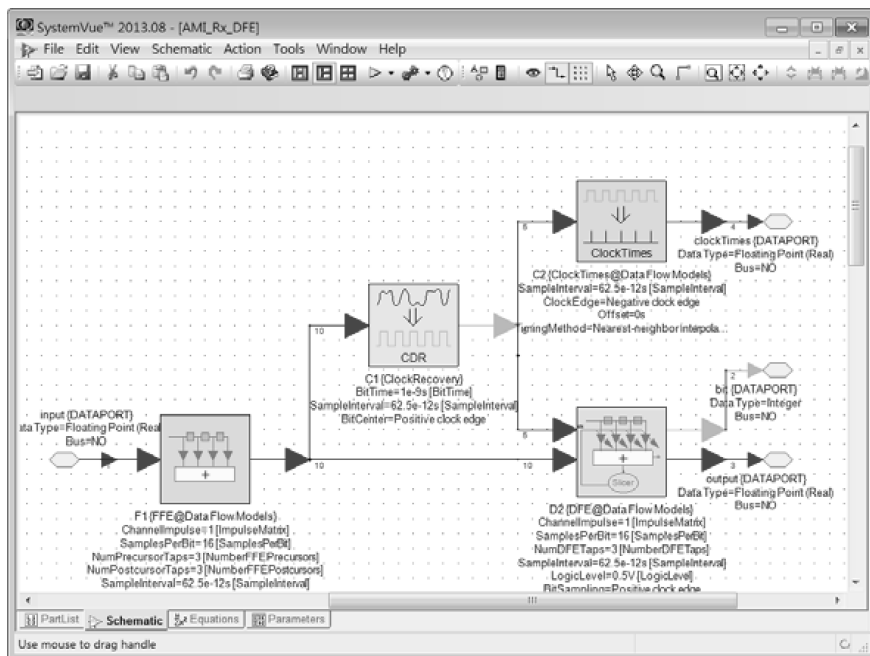


图 A-15 System Vue 中 SERDES 接收端建模实例

SystemVue 中 SERDES 接收机建模的实例。接收机模型包括 FFE、CDR 和 DFE 均衡器。这些组件的参数是可编辑的。我们可以修改这些组件的参数并执行数据流量仿真，检查系统性能，如眼图等。此外，模型可以以 AMI 格式导出。然后，我们可以在在 ADS 中使用这些 IBIS AMI 模型进行通道的仿真。

SystemVue 提供了 AMI 收发模型库用于 SERDES 架构工程师 AMI 模型，如图 A-16 所示。模型库包括 SERDES 系统建模所需的元件，如 VCO、FFE、DFE 等，我们也可以通过 C++ 语言生成特定元件并在 SystemVue 中使用它们。



图 A-16 SystemVue 中的 AMI 模型库

若要创建 AMI 模型：

- (1) 使用 AMI 库元件建立如图 A-16 所示的 Tx 和 Rx 模型。
- (2) 执行数据流量仿真，确认模型符合设计目标。
- (3) 在如图 A-17 所示的窗口生成和编译 C++ 代码，产生 AMI 兼容的 .dll 和 .ami 文件。

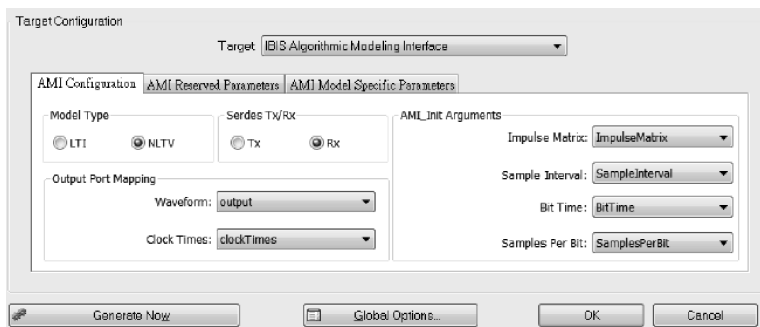


图 A-17 AMI 模型生成窗口

- (4) 将 .ibs、.dll 文件和 .ami 文件放在相同的文件夹中，在 .ibs 文件中使用关键字 [AlgorithmicModel] 链接相关的文件。
- (5) 在 ADS 中验证模型，如图 A-18 所示，确保所有参数都可用，性能满足要求。
- (6) 打包发布 IBIS AMI 模型。

使用 System Vue 建立 AMI 模型的实例请参考实例 9。

**【问题 9】** 如何建立高速通道模型？建模时要注意哪些问题？

在高速数字信号传输中，通道是从发送端的 IO buffer 到接收端 IO buffer 的电气路径。它可能包括集成电路封装、PCB 走线、连接器和电缆等。图 A-19 显示了一个复杂高速背板

的通道构成。

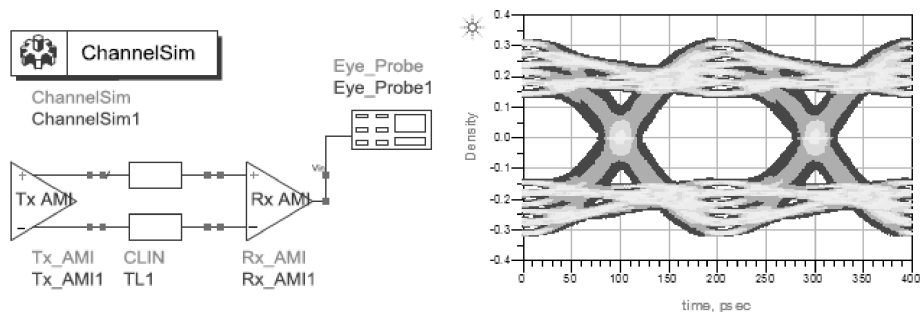


图 A-18 AMI 模型在 ADS 中进行验证

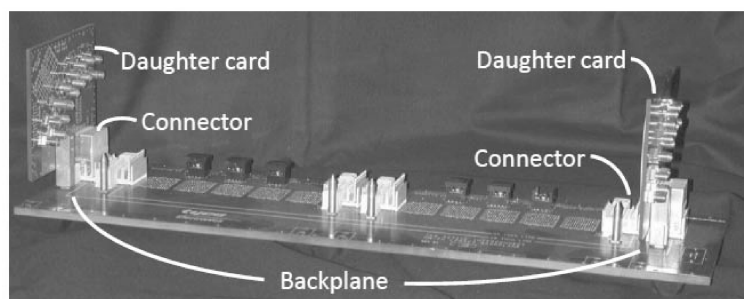


图 A-19 高速背板的通道构成

通道建模必须注意三个最重要的性能，即阻抗、损耗、相位或延迟。

阻抗取决于几何结构、介电常数和金属的电导率。阻抗不匹配会导致多重反射，它会导致信号振铃、过冲和下冲。损耗是通道设计的关键，通道损耗的主要来源是介质损耗和导体损耗，它一般与频率相关。通道的带宽限制最大比特率。对 DDR 设计，必须很好地控制命令/地址和时钟及 DQ 和 DQs 之间的相位差。

通道的每个组件可以由传输线模型、电磁仿真模型或测量模型组成。每个组件的正确建模依赖于精确的尺寸和材料参数。当模型定义好后，级联这些组件来构建一个完整的通道模型，如图 A-20 所示。然后，由 TDR 仿真，如图 A-21 所示，可以估计阻抗和延迟特性。通道的损耗分析可以通过 S 参数仿真获得，如图 A-22 所示。

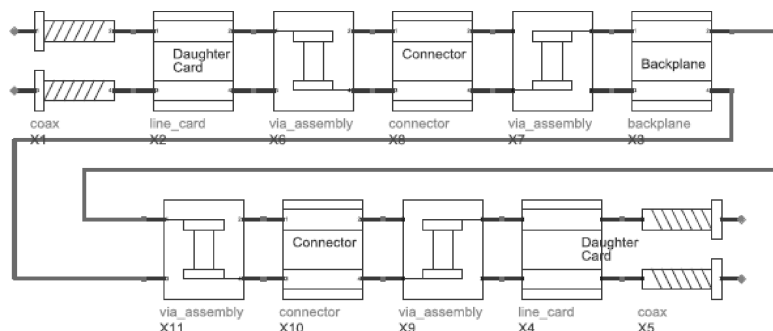


图 A-20 完整的通道模型

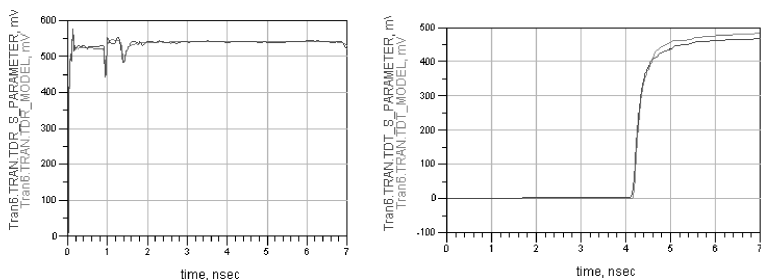


图 A-21 TDR 仿真通道的阻抗与延迟特性

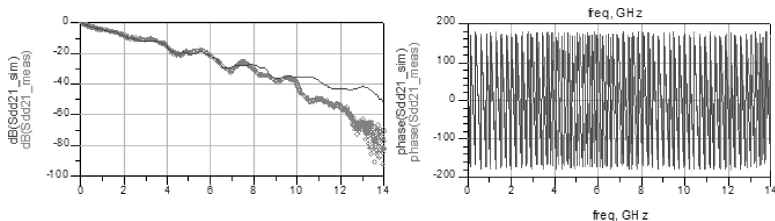


图 A-22 插入损耗与相位

【问题 10】如何设计阻抗控制的孔与通过绘图实用程序？

过孔的阻抗控制是影响通道性能的关键。要达到较好的阻抗控制，孔径、焊盘、反焊盘和过孔的间距必须精心设计。如图 A-23 所示，过孔绘图工具 Via Drawing Utility 是 ADS 中建

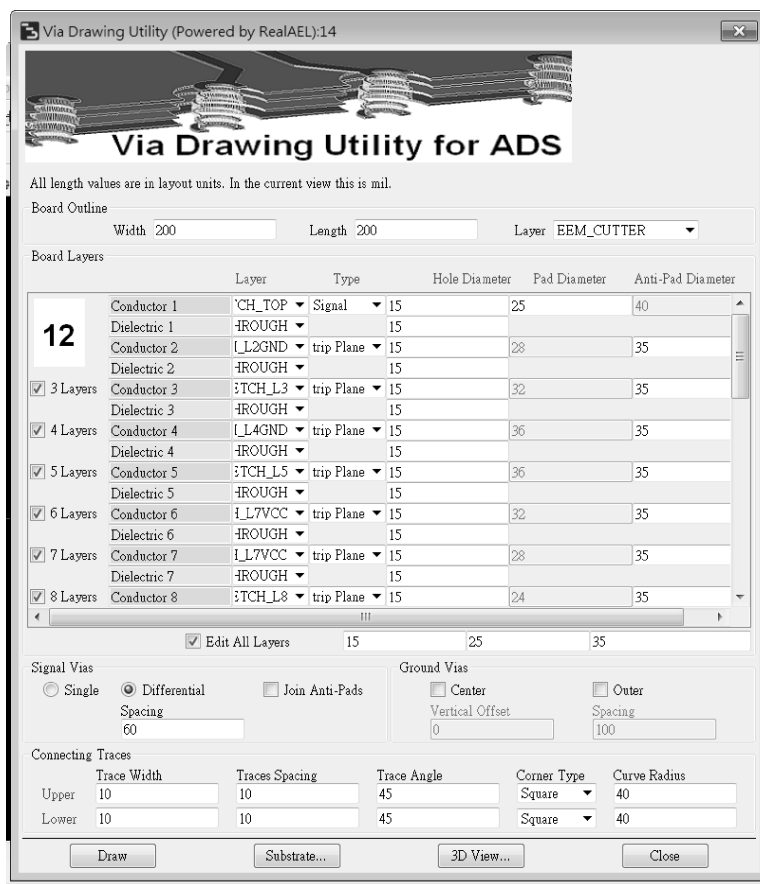


图 A-23 过孔绘图工具 Via Drawing Utility

造复杂过孔结构的一个小工具。我们可以在每个图层上输入间距、开孔直径、焊盘和反焊盘尺寸等参数，然后单击绘制按钮来生成孔的布局，如图 A-24 所示。

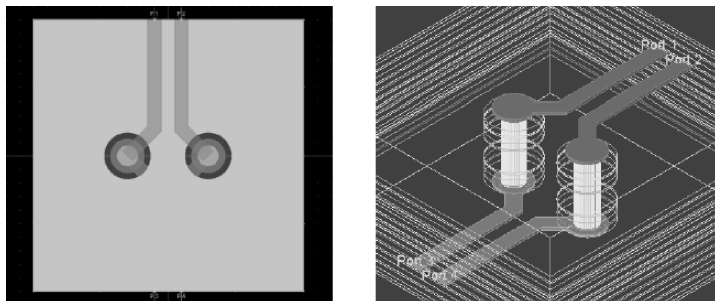


图 A-24 过孔对的版图与三维结构图

可以在 ADS 中用有限元法 (FEM) 或矩量法 (MoM) 仿真提取一对过孔的 S 参数，然后用 TDR 仿真检查过孔的阻抗，如图 A-25 所示。如图 A-26 所示，TDR 结果表明阻抗不匹配。通过过孔绘图工具中的结构调整再仿真，实现阻抗匹配过孔设计。

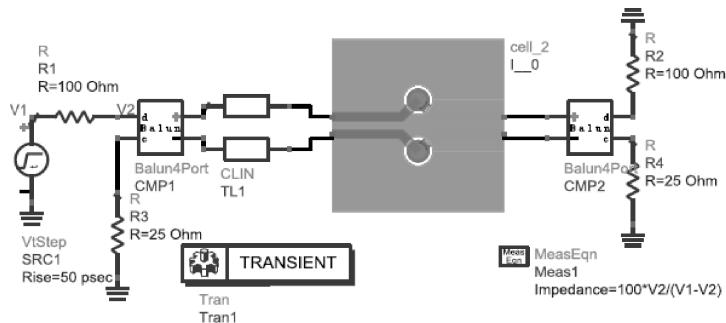


图 A-25 过孔对的 TDR 仿真

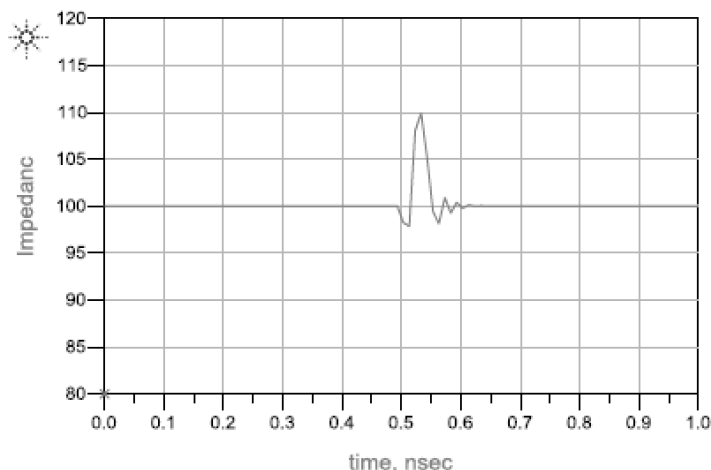


图 A-26 TDR 仿真的阻抗结果

【问题 11】宽带 SPICE 模型是什么，如何生成？

高速数字设计中，S 参数模型是通道与通道间的串扰分析的常用模型。ADS 瞬态仿真器

带有先进的卷积仿真引擎，用户可以直接使用 S 参数作为瞬态仿真中的信道模型。

然而，对于设计师来说，他们使用的 SPICE 工具可能不容易处理 S 参数。当 S 参数频带很宽、端口数较多时，仿真结果可能是不准确的。为了给 SPICE 工具中的瞬态仿真提供准确和稳定的模型，ADS 提供了宽带 SPICE 模型生成器，将 S 参数转换为其等效的宽带 SPICE 模型。

要启动宽带 SPICE 模型生成器，请选择 Tools→SPICE Model Generator→Start Broadband Generator。宽带 SPICE 模型生成器界面如图 A-27 所示。我们可以选择输入 Touchstone 或 .ds 格式的 S 参数文件，然后选择输出 SPICE 格式导出。支持的 SPICE 格式包括 HSPICE™、SPECTRE™、SPICE2 和 SPICE3。

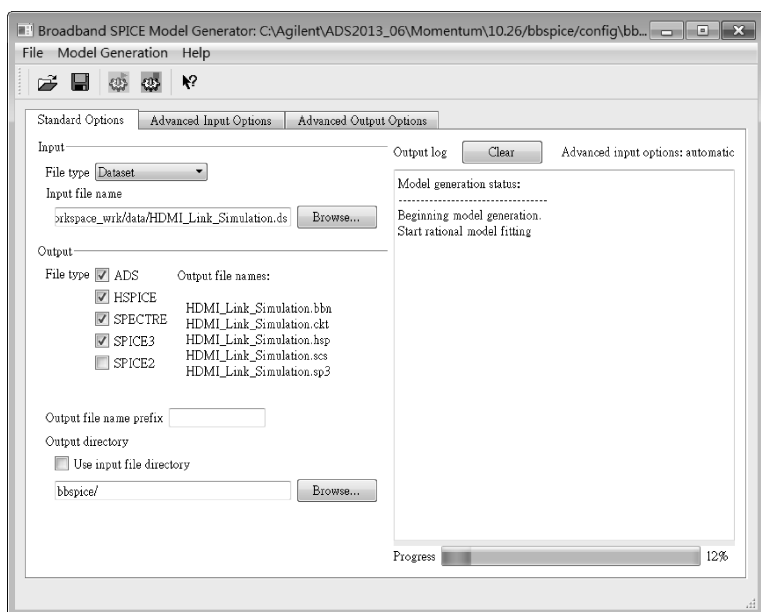


图 A-27 宽带 SPICE 模型生成器界面

生成的 SPICE 文件将保存在工作区下的“bb spice”文件夹中。YONGHU 可以在瞬态仿真中使用等效的 SPICE 模型。数据显示窗口将显示输入的 S 参数数据和其等效电路的宽带 SPICE 模型的 S 参数结果的差异，帮助用户检查 SPICE 模型的质量，如图 A-28 所示。

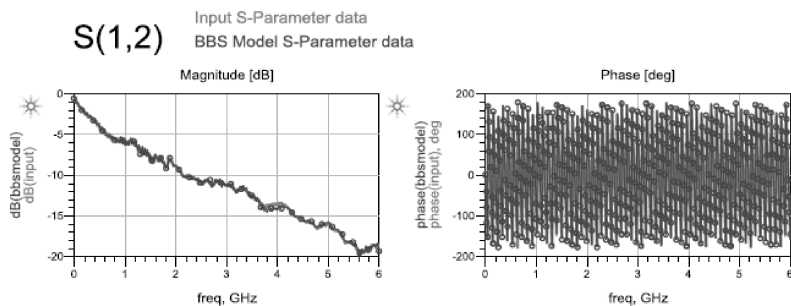


图 A-28 输入 S 参数与等效 SPICE 模型的 S 参数比较

【问题 12】如何从 S 参数提取简单的 RLC 等效电路？



可以用 ADS 将 S 参数模型转换为其等效的 RLC 电路。例如，IC 封装通常用 RLC 电路描述其特征。图 A-29 左边显示了一个简化的封装结构；图 A-29 右显示芯片封装的等效电路的拓扑结构。

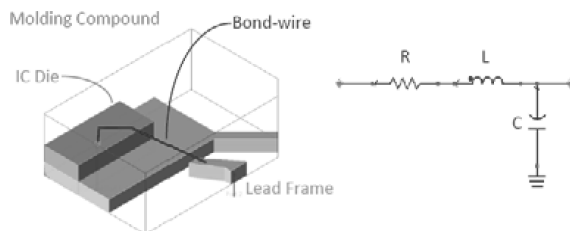


图 A-29 简化的封装结构（左）和它的等效 RLC 电路（右）

若要提取 R、L 和 C，一个简单的方法是使用下面的等式：

$$R = 1/\text{real}(Y_{11})$$

$$L = \text{imag}(1/Y_{11})/(2 * \pi * \text{freq})$$

$$C = \text{imag}(1/Z_{22})/(2 * \pi * \text{freq})$$

由以上等式，可以得到 R、L 和 C，如图 A-30 所示。

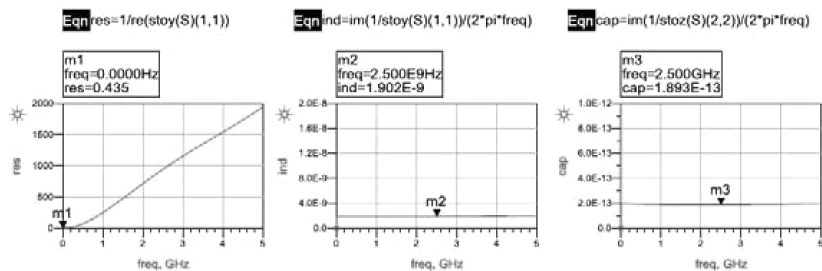


图 A-30 提取 R、L、C 与 ADS 方程

比较原有的 S 参数模型和其等效电路的 S 参数，如图 A-31 所示。这两个结果都非常接近，如图 A-32 所示。

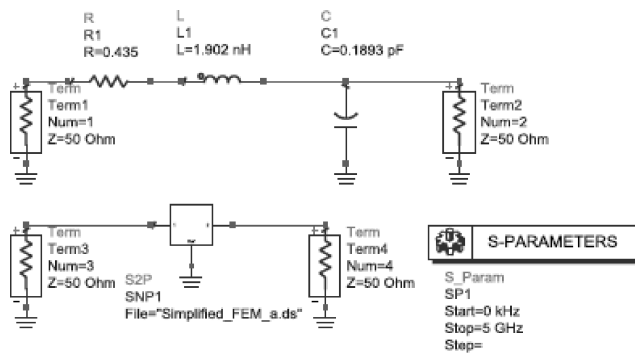


图 A-31 S 参数和等效 RLC 的比较

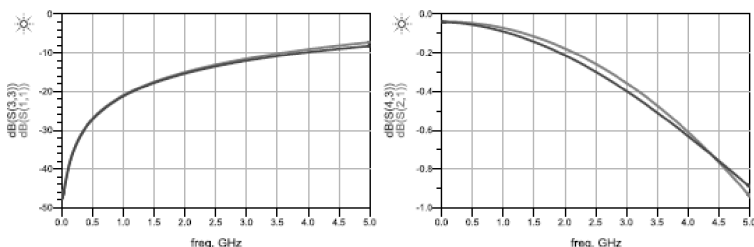


图 A-32 S 参数和等效 RLC 电路的结果

【问题 13】如何设置 ADS 信号源元件中的比特序列？

在 ADS 常用的时域信号源 ViPRBS 中，有四种模式用于生成不同的比特序列：

- ☺ 最大长度线性反馈移位寄存器 (Maximal Length LFSR)；
- ☺ 用户定义线性反馈移位寄存器 (User Defined LFSR)；
- ☺ 比特序列 (Explicit Bit Sequence)；
- ☺ 外部比特文件 (Bit File)。

在模式 1 和模式 2 中，使用线性反馈移位寄存器 (LFSR) 生成伪随机二进制序列 (PRBS)。在模式 3 下，我们可以输入任意的比特序列。在模式 4 下，信号源读取文本格式的比特序列文件。

图 A-33 显示一个三抽头线性反馈移位寄存器的例子。在此示例中，Tap3 和 Tap1 是异或 “XOR” 运算的输入端，标记为 “101”。寄存器的初始值称为种子。在此示例中，种子是 “011”。每个时钟周期，在寄存器中的值右移，Tap1 为 XOR 的输出值。Tap3 的输出是线性反馈移位寄存器生成的比特序列。

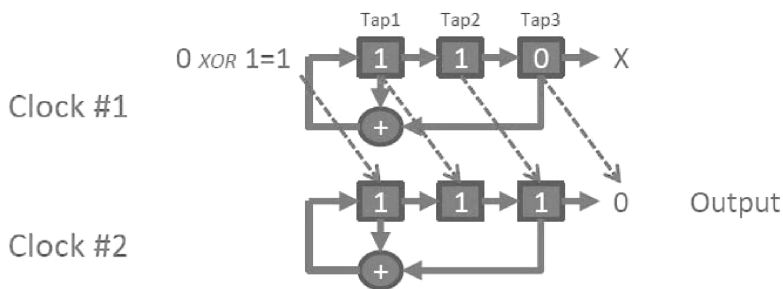


图 A-33 三抽头线性反馈移位寄存器

理论上，比特序列的最大周期长度是  $2^N - 1$ ，其中  $N$  是寄存器长度。这意味着比特序列将在  $2^N - 1$  个时钟周期后重复。图 A-34 显示了三抽头 LFSR 流动状态。在此示例中，比特序列 “0111010” 每  $2^3 - 1 = 7$  个时钟周期重复一次。在模式 1 中，默认寄存器的长度是 8，这意味着比特序列将在  $2^8 - 1 = 255$  个时钟周期后重复。若要获得较长周期长度的比特序列，我们可以设置更大的寄存器长度。例如，如果寄存器长度设置为 31，最大的比特序列周期长度将为  $2^{31} - 1 = 2147483647$ 。在模式 2 中，我们可以输入抽头系数和种子值来生成特定的 PRBS，如图 A-35 所示。

【问题 14】如何执行批处理模拟有效与批处理仿真控制器？

要了解电路性能如何随电路参数变化，常用的方法是对电路设计中的参数进行扫描。批处理仿真控制器是一个简单易用的小工具，可以帮助用户建立复杂的扫描过程。如图 A-36

所示。它支持“Use sweep plan”和“Use sweep module”两种模式。在“Use sweep plan”模式中，可以输入变量扫描。扫描范围在如图 A-37 所示的批处理仿真对话框中设置。

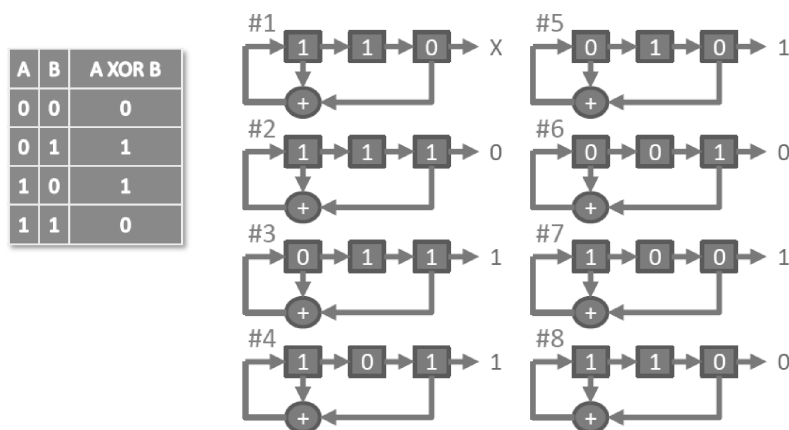


图 A-34 3 抽头线性移位寄存器状态图

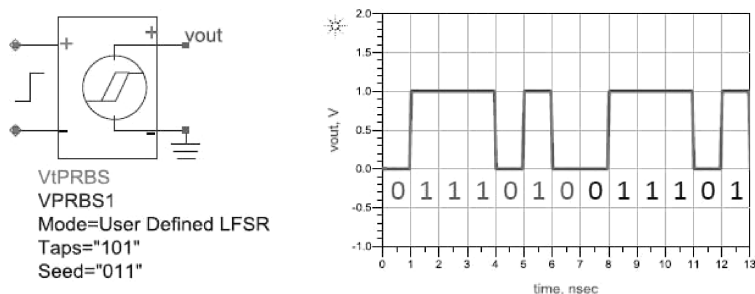


图 A-35 用户定义 LFSR 模式

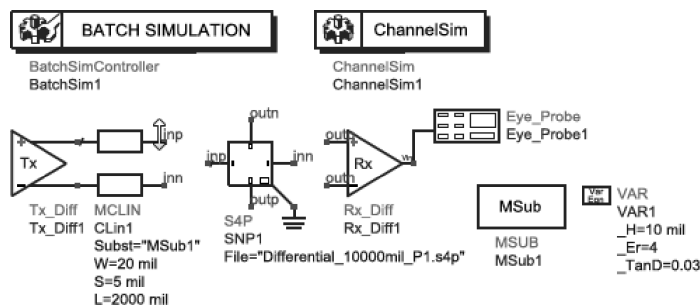


图 A-36 批处理仿真原理图

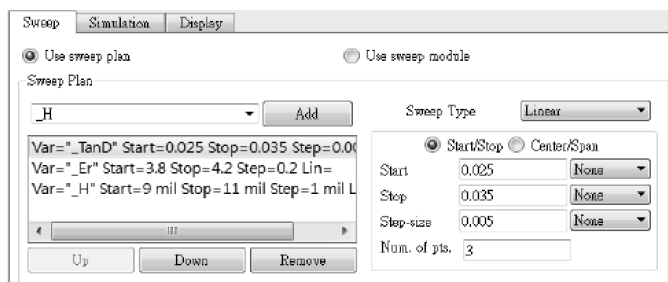


图 A-37 批处理仿真对话框

仿真结束时，可以在一个矩形框图中显示所有仿真运行的结果，如图 A-38 所示。我们可以轻松地检查电路性能的变化。然而，文件名和字符串是不支持“Use sweep plan”模式直接扫描的。解决方案是建立一个 DataFileList 组件的文件列表（或 StringList 组件中的字符串列表），如图 A-39 所示，生成文件名与指针之间的映射表，然后在“Use sweep plan”模式下扫描指针。

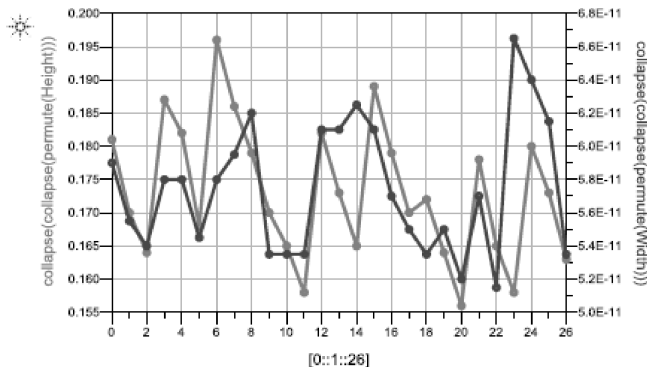


图 A-38 眼高和眼宽与参数的变化关系

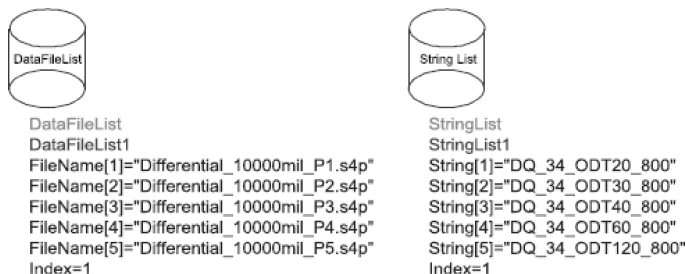


图 A-39 DataFileList 或 StringList 的组件

在“Use sweep module”模式下，我们用文本编辑器可以建立一个 .csv 文件定义扫描计划，如图 A-40 所示。此模式支持数值、文件名和字符串。仿真开始时，批处理仿真控制器将基于该文件中列出的项目设置变量，依次执行仿真，直到所有的项目仿真完毕。

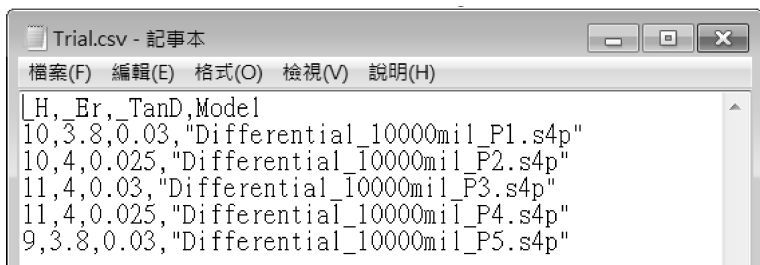


图 A-40 在“Use sweep module”模式下使用 .csv 文件

#### 【问题 15】如何计算混合模式 S 参数的插入损耗、回波损耗和串扰？

混合模式 S 参数通常用于检查差分电路的频域特性，包括插入损耗、回波损耗和平衡非平衡模式的转换。通常情况下，我们可以用等式将四个端口单端的 S 参数转换为混合模式 S 参数，如图 A-41 所示；或者，可以使用巴伦电路进行转换，如图 A-42 所示。混合模式 S

参数仿真结果如图 A-43 所示。

**Mixed-Mode S-Matrix**  
(derived from single-ended S-Matrix)

Eqn SDD11=0.5*(S(1,1)-S(1,3)-S(3,1)+S(3,3))	Eqn SDC11=0.5*(S(1,1)+S(1,3)-S(3,1)-S(3,3))
Eqn SDD21=0.5*(S(2,1)-S(2,3)-S(4,1)+S(4,3))	Eqn SDC21=0.5*(S(2,1)+S(2,3)-S(4,1)-S(4,3))
Eqn SDD12=0.5*(S(1,2)-S(1,4)-S(3,2)+S(3,4))	Eqn SDC12=0.5*(S(1,2)+S(1,4)-S(3,2)-S(3,4))
Eqn SDD22=0.5*(S(2,2)-S(2,4)-S(4,2)+S(4,4))	Eqn SDC22=0.5*(S(2,2)+S(2,4)-S(4,2)-S(4,4))
Eqn SCC11=0.5*(S(1,1)-S(1,3)+S(3,1)-S(3,3))	Eqn SCC11=0.5*(S(1,1)+S(1,3)+S(3,1)+S(3,3))
Eqn SCC21=0.5*(S(2,1)-S(2,3)+S(4,1)-S(4,3))	Eqn SCC21=0.5*(S(2,1)+S(2,3)+S(4,1)+S(4,3))
Eqn SCC12=0.5*(S(1,2)-S(1,4)+S(3,2)-S(3,4))	Eqn SCC12=0.5*(S(1,2)+S(1,4)+S(3,2)+S(3,4))
Eqn SCC22=0.5*(S(2,2)-S(2,4)+S(4,2)-S(4,4))	Eqn SCC22=0.5*(S(2,2)+S(2,4)+S(4,2)+S(4,4))

图 A-41 混合模式 S 参数计算公式

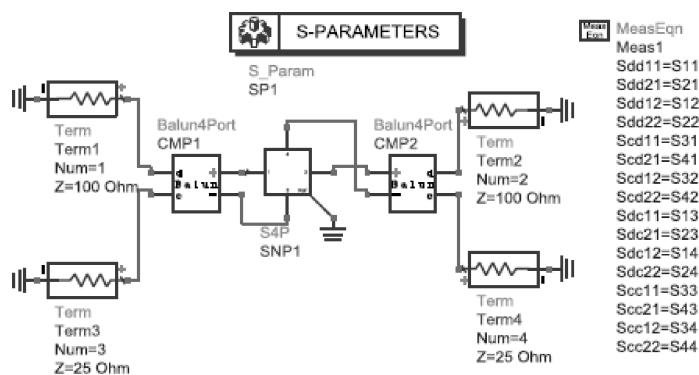


图 A-42 混合模式 S 参数仿真原理图

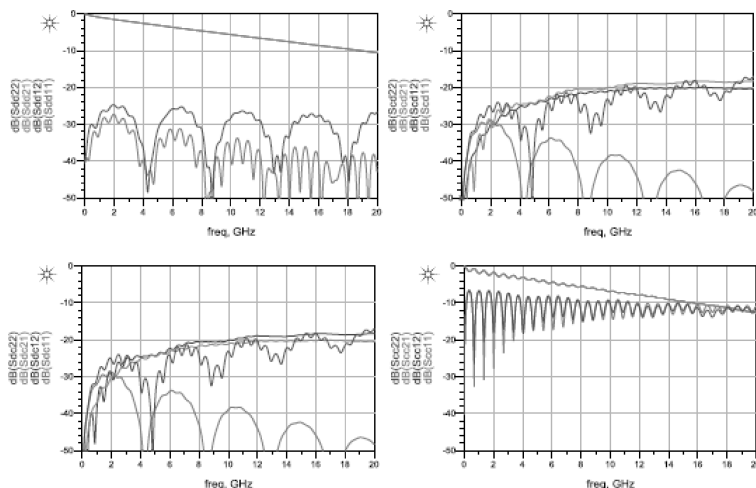


图 A-43 混合模式 S 参数仿真结果

当差分对数量较多时，计算差模对之间的串扰公式将非常复杂。我们可以使用如图 A-44 所示的 S 参数仿真来简化工作。例如，S(5,1)是左侧上端的差分信号对左侧下端差分信号的近端串扰（NEXT），S(6,1)是左侧上端差分信号对右侧下端差分信号的远端串扰（FEXT）。

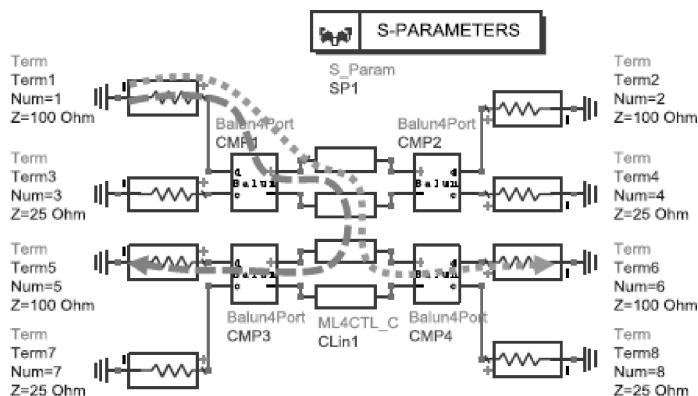


图 A-44 (a) 混合模式 S 参数中的串扰仿真

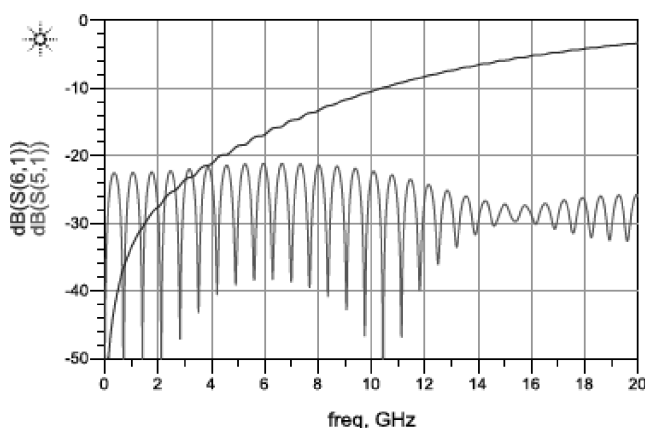


图 A-44 (b) 混合模式 S 参数中的串扰

【问题 16】在 ADS 中如何由 S 参数得到 TDR 阻抗结果？

ADS 支持两种方式将通道的 S 参数转换为 TDR 阻抗。一种方式是由瞬态仿真获得，如图 A-45 所示，通道的阻抗变化可以由电压源内阻两端的电压 V1 和 V2 计算得到： $Z = Z_0 \times V_2 / (V_1 - V_2)$ 。其中， $Z_0$  是电压源阻抗。

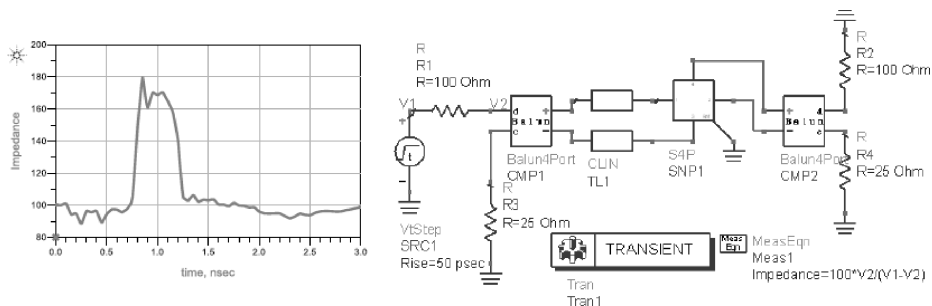


图 A-45 TDR 的瞬态仿真

另一种方式是通过数据显示窗口中的 SP - TDR 面板工具获得，如图 A-46 所示。SP TDR 界面类似矢量网络分析仪，它通过逆傅里叶变换由频域数据计算端口的阶跃响应获取时域阻抗结果。左边的面板按钮提供 TDR 计算的各种功能。例如，“Peeling”功能可以减轻

多重反射所致的阻抗不匹配，以得到更准确的阻抗；“Smooth”功能可以滤除测量噪声，以提高信号质量。

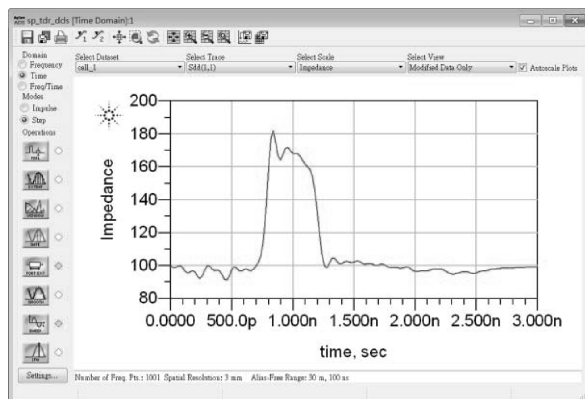


图 A-46 SP-TDR 面板工具

【问题 17】能否用 ADS 进行抖动分离？

Keysight 数字示波器中的 EZJIT + 是进行误码率 (BER) 估计、抖动分析的有力工具。EZJIT + 能够分离不同类型的抖动，包括随机抖动 (RJ)、周期抖动 (PJ) 和数据相关抖动 (DDJ) 等。在 EZJIT + 中我们可以检查抖动直方图和每种抖动类型的统计结果，如图 A-47 所示。



图 A-47 EZJIT + 抖动分析

ADS 数据显示窗口中的抖动分析前面板工具采用了与 EZJIT + 相同的抖动分离技术。它可以在设计阶段执行完整的抖动分析，如图 A-48 所示。

抖动分析前面板还支持抖动频谱。抖动频谱显示总抖动的频谱信息。它对找出抖动来源很有帮助。例如，如果我们怀疑抖动源是一个开关频率 66MHz 的信号，可以通过检查抖动频谱是否在 66MHz 存在峰值来验证这一假设，如图 A-49 所示。

【问题 18】如何从 ADS 导出波形文件到 Infiniium 软件，进行眼图和抖动分析？

Keysight 的 Infiniium offline 是一款功能强大的示波器分析软件。它可以从 ADS 中读取波形并执行各种数字信号分析。若要在 ADS 中生成 Infiniium 兼容波形文件，需要在原理图中放置 Infiniium Binary File Output 元件，如图 A-48 所示，然后运行通道仿真计算接收端的信

号和时钟波形。仿真结束时，将生成两个 infinium 兼容二进制文件，ChannelSim1\_Waveform\_Eye\_Probe1. bin、ChannelSim1\_ClockSignal\_Eye\_Probe1. bin，如图 A-50 所示。在 Infimum 软件中，我们可以读这两个文件，并检查波形和时钟文件，如图 A-51 所示。

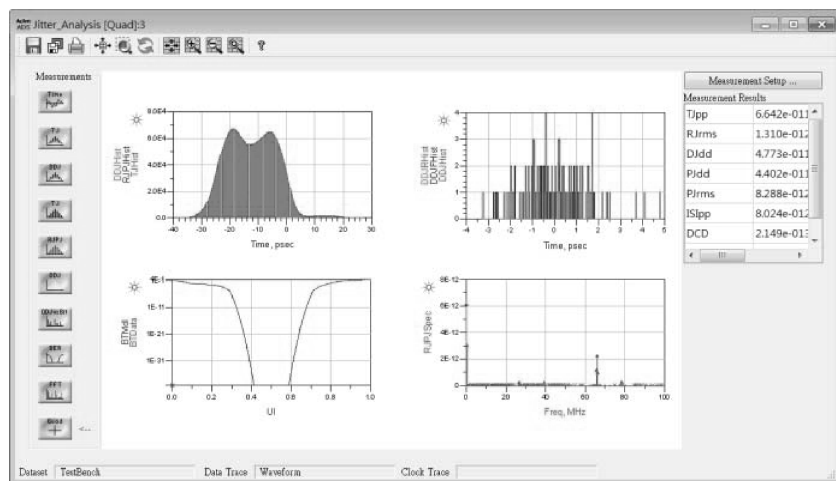


图 A-48 ADS 抖动分析前面板

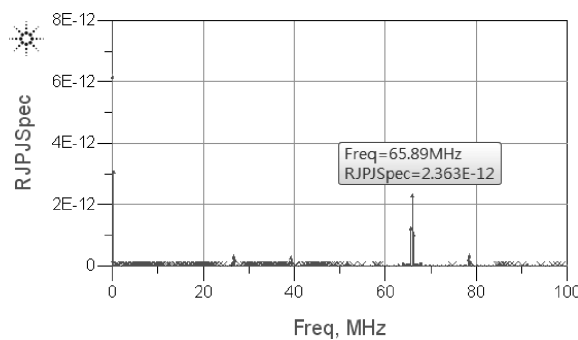


图 A-49 抖动频谱显示峰值为 66MHz

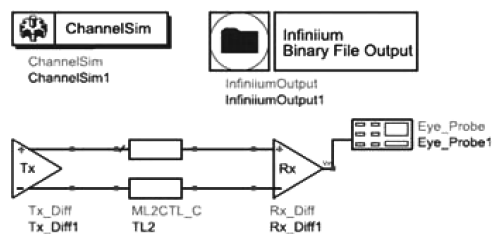


图 A-50 输出 Infimum 兼容波形和时钟文件



图 A-51 Infimum 读入 ADS 仿真的波形和时钟文件



Infiniium 为高速数字信号包括眼统计、抖动分离和符合性测试提供了有力的分析工具。例如，我们可以查看基于仿真波形的眼图，如图 A-52 所示。或者，我们可以用 EZJIT + 得到抖动的统计数据，如抖动分离、抖动直方图、抖动频谱和误码率浴盆曲线，如图 A-53 所示。

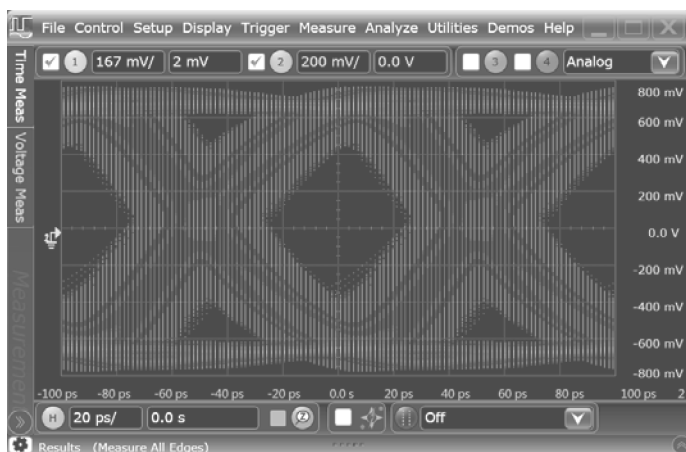


图 A-52 串行数据分析构建眼图

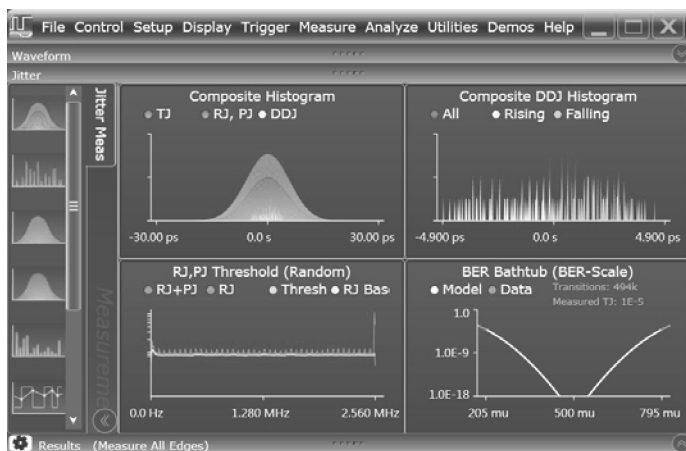


图 A-53 EZJIT + 抖动分析

#### 【问题 19】发射端芯片预加重的原理是什么？

当高速数字信号通过一个有损的通道时，由于高频信号分量的损失，信号边缘会产生恶化，如图 A-54 右图所示，它导致码间干扰 (ISI) 和眼图闭合。为了减轻高频损失，一种方法是在发射端 (Tx) 使用预加重电路。预加重电路在每一次电平转换时都提高高频能量，如图 A-55 左图所示，从而补偿接收端 (Rx) 信号的高频损失，提高信号质量，如图 55 右图中的所示。

预加重电路可以由多抽头有限脉冲响应 (FIR) 滤波器构成，如图 A-56 所示。我们可以通过设置抽头系数 (C0、C1、C2) 调整滤波器响应。在 ADS 中，Tx\_Diff 和 Tx\_SingleEnded 元件允许用户输入抽头系数，如图 A-57 所示。图 A-58 显示了关闭和开启预加重后的眼图比较，可以看到经过预加重后眼图明显张开。

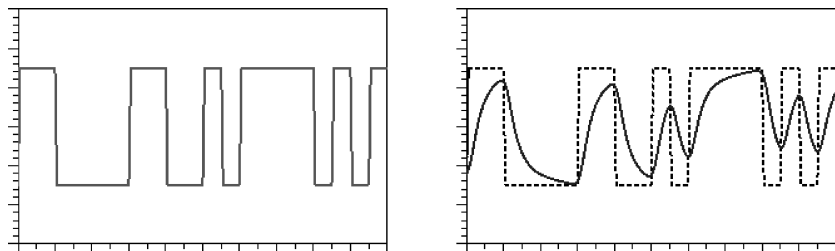


图 A-54 高频损失造成 RX 信号恶化

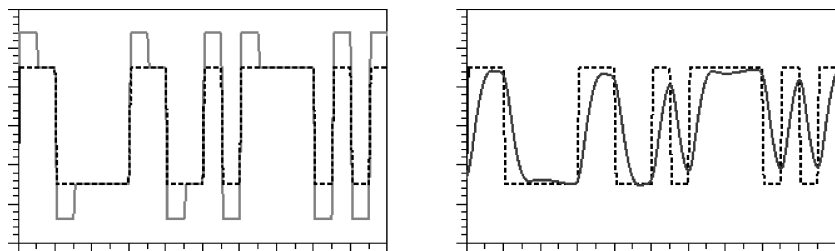


图 A-55 预加重补偿高频损失，提高信号质量

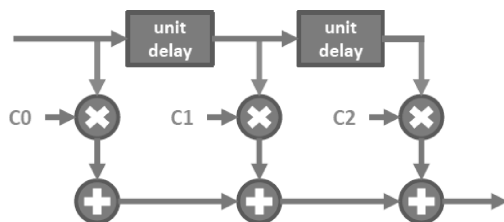


图 A-56 三抽头预加重电路结构

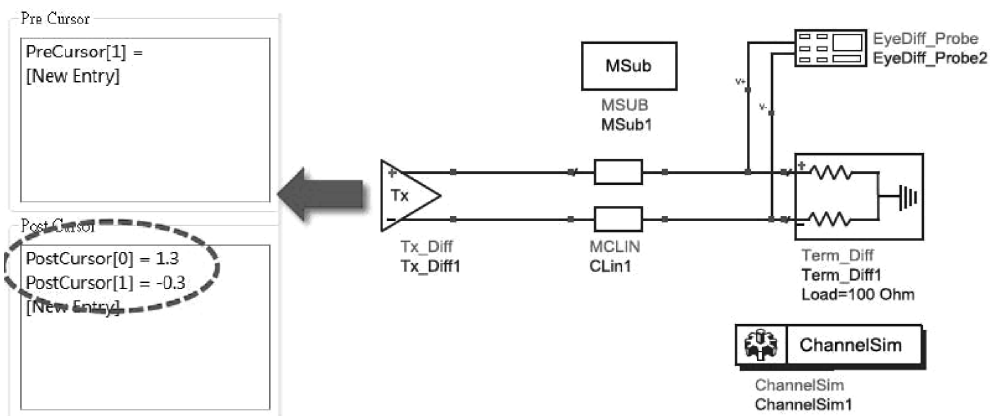


图 A-57 在 Tx\_Diff 组件中设置抽头系数

【问题 20】连续时间线性均衡器（CTLE）如何改善高速数字信号质量？

CTLE 是接收端（Rx）模拟滤波器，由有源或无源电路构成。当高速数字信号通过有损通道传播时，接收端 CTLE 通过提高高频分量的信号来补偿高频通道损失，并通过 CTLE 传递函数查看其频域行为。例如，USB3.0 CTLE 长通道的传递函数为

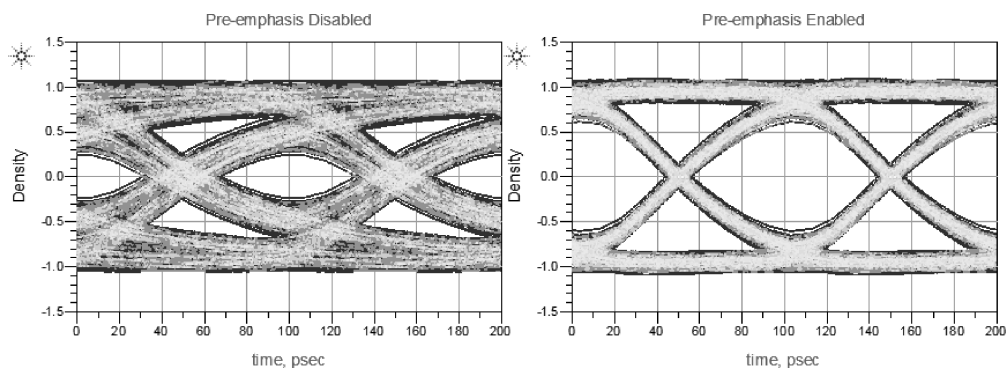


图 A-58 预加重关闭（左）和开启（右）后的眼图

$$H(s) = \frac{A_{dc} \omega_{p1} \omega_{p2}}{\omega_z} \cdot \frac{s + \omega_z}{(s + \omega_{p1})(s + \omega_{p2})}$$

式中,  $A_{dc} = 0.667$ ;  $\omega_z = 2\pi \times 650\text{MHz}$ ;  $\omega_{p1} = 2\pi \times 1.95\text{GHz}$ ;  $\omega_{p2} = 2\pi \times 5\text{GHz}$ 。

图 A-59 所示的是 CTLE 的传递函数。图 A-60 所示的是 CTLE 的阶跃响应。我们可以看到在上升沿边缘信号电平由于 CTLE 引起的放大。在 ADS 中, Rx\_Diff 和 Rx\_SingleEnded 的组件允许用户输入 CTLE 传递函数的零点、极点, 如图 A-61 所示的。图 A-62 所示为关闭和开启 CTLE 后的眼图比较。我们可以看到经过预加重后眼图明显张开。

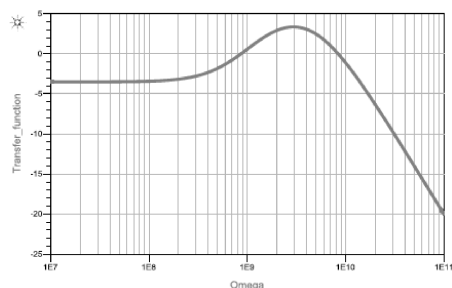


图 A-59 USB3.0 CTLE 长通道的传递函数

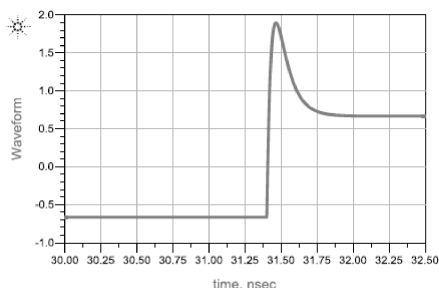


图 A-60 CTLE 的阶跃响应

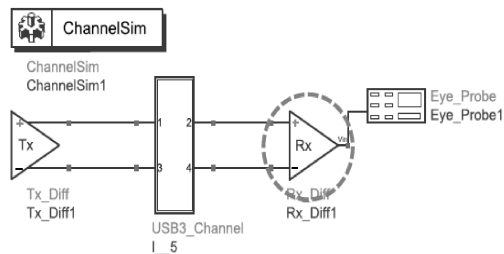
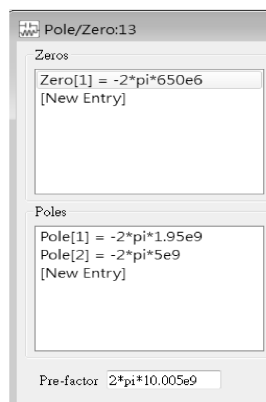


图 A-61 设置 CTLE 传递函数的零点、极点



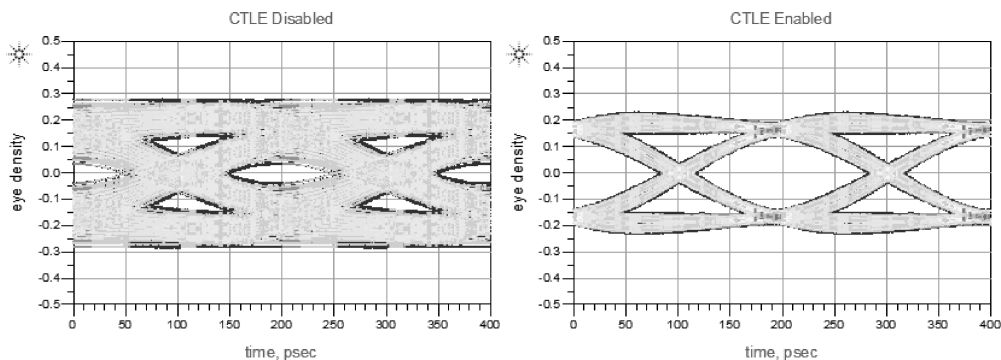


图 A-62 CTLE 关闭 (左) 和 CTLE 开启 (右) 的眼图比较

【问题 21】前馈均衡器 (FFE) 如何改善高速数字信号质量?

FFE 的工作原理与问题 19 介绍的预加重类似。FFE 用于 SERDES 接收端 (Rx) 减轻小符号间干扰 (ISI) 对通道性能的影响。图 A-63 显示了三抽头 FFE 示例。抽头系数  $C_0$ 、 $C_1$ 、 $C_2$  经过了优化, 以尽量减少码间干扰。

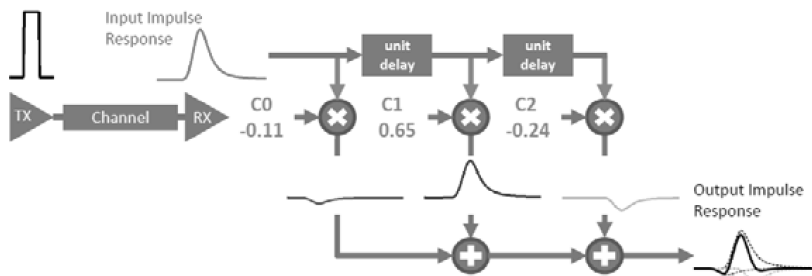


图 A-63 三抽头 FFE 示例

【问题 22】判决反馈均衡 (DFE) 如何改善高速数字信号质量?

DFE 通常用于 SERDES 接收端 (Rx) 以减小符号间干扰 (ISI)。DFE 由有限脉冲响应 (FIR) 滤波器、加法器和限幅器 (用于符号判决) 组成。图 A-64 显示了三抽头 DFE 示例。DFE 的优点是可以降低符号间干扰又不会放大噪声, 它的缺点是不能纠正前标记 (precursor) 符号间干扰。

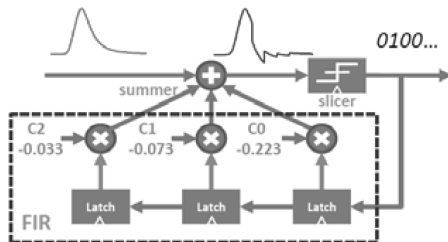


图 A-64 三抽头 DFE 示例

首先, FIR 滤波器计算前一符号的符号间干扰 (ISI); 然后, 干扰将从接收到的信号中减去; 最后, 限幅器进行符号判决, 将符号输出到 FIR 滤波器。在此示例中, 抽头系数是  $-0.223$ 、 $-0.073$  和  $-0.033$ , 如图 A-65 所示, 在采样点处前一符号的干扰被消除。

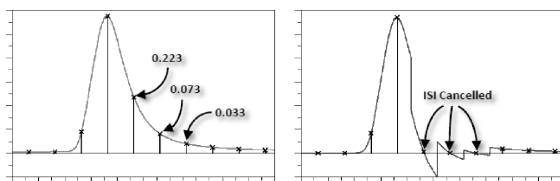


图 A-65 输入脉冲（左）、输出脉冲响应（右）

DFE 的通道仿真原理图如图 A-66 所示。Rx\_SingleEnded 和 Rx\_Diff 元件允许用户启用“optimized initial tap calculations”计算 DFE 抽头系数（见图 A-67）。仿真结束时，最佳的抽头系数将写入到一个文本文件中。图 A-68 显示了关闭和开启 DFE 后的眼图比较。我们可以看到经过 DFE 信号后质量得到明显改善。

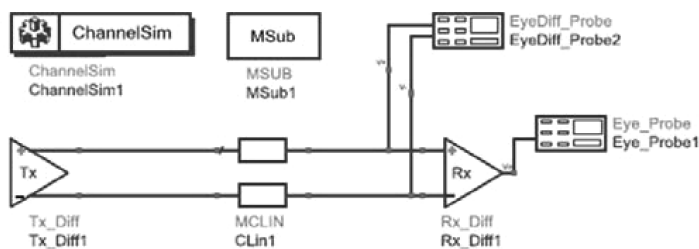


图 A-66 通道仿真原理图

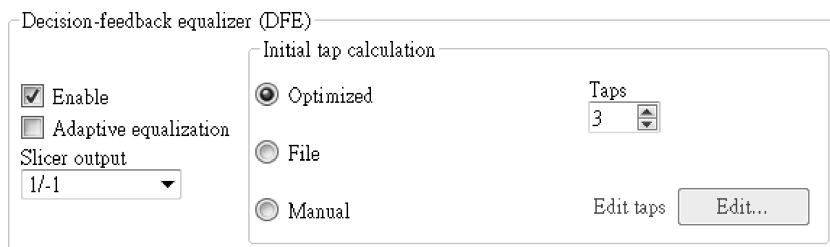


图 A-67 启用优化 DFE 抽头系数

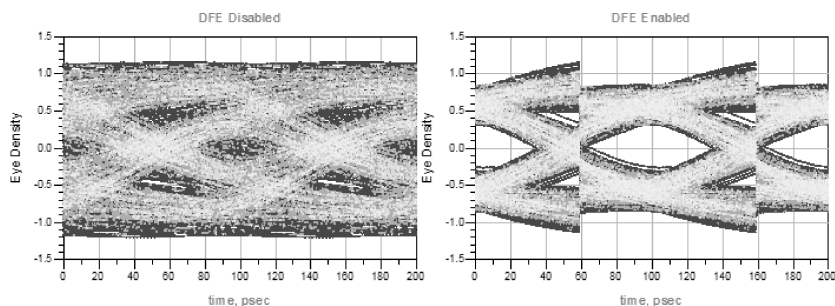


图 A-68 DFE 关闭（左）与 DFE 启用（右）的眼图比较

# 反侵权盗版声明

电子工业出版社依法对本作品享有专有出版权。任何未经权利人书面许可，复制、销售或通过信息网络传播本作品的行为；歪曲、篡改、剽窃本作品的行为，均违反《中华人民共和国著作权法》，其行为人应承担相应的民事责任和行政责任，构成犯罪的，将被依法追究刑事责任。

为了维护市场秩序，保护权利人的合法权益，本社将依法查处和打击侵权盗版的单位和个人。欢迎社会各界人士积极举报侵权盗版行为，本社将奖励举报有功人员，并保证举报人的信息不被泄露。

举报电话：(010) 88254396；(010) 88258888

传 真：(010) 88254397

E-mail: dbqq@phei.com.cn

通信地址：北京市海淀区万寿路173信箱

电子工业出版社总编办公室

邮 编：100036